Robust Median Filtering Forensics Using an Autoregressive Model

Xiangui Kang, *Member, IEEE*, Matthew C. Stamm, *Member, IEEE*, Anjie Peng, and K. J. Ray Liu, *Fellow, IEEE*

***Abstract*—In order to verify the authenticity of digital images, researchers have begun developing digital forensic techniques to identify image editing. One editing operation that has recently received increased attention is median filtering. While several median filtering detection techniques have recently been developed, their performance is degraded by JPEG compression. These techniques suffer similar degradations in performance when a small window of the image is analyzed, as is done in localized filtering or cut-and-paste detection, rather than the image as a whole. In this paper, we propose a new, robust median filtering forensic technique. It operates by analyzing the statistical properties of the median filter residual (MFR), which we define as the difference between an image in question and a median filtered version of itself. To capture the statistical properties of the MFR, we fit it to an autoregressive (AR) model. We then use the AR coefficients as features for median filter detection. We test the effectiveness of our proposed median filter detection techniques through a series of experiments. These results show that our proposed forensic technique can achieve important performance gains over existing methods, particularly at low false-positive rates, with a very small dimension of features.**

***Index Terms*—Median filtering, noise residual, image forensics, autoregressive model.**

I. Introduction

BECAUSE digital images can be easily edited, it is often difficult to tell if a digital image has been manipulated. To combat this problem, researchers have developed a variety of blind forensic techniques to verify the authenticity of digital images [1]–[6], [8], [11], [12], [15]–[17], [21], [22]. Many of these techniques operate by searching for imperceptible traces, known as fingerprints, that are introduced into an image by editing operations. By identifying these fingerprints, a forensic investigator can determine if and how an image was manipulated. A number of forensic techniques [22] currently exist to detect the use of resampling [5], contrast enhancement [6], multiple compression [15], [16], sharpening [21], and blurring [22].

One image editing operation that has received increased attention from digital forensic researchers is median filtering [1]–[4], [25]. Median filtering is a nonlinear operation that has the useful property of preserving edges within an image. It is commonly used to perform image denoising, remove outlying pixel values, and to smooth regions of an image. Because of this, forgers may use median filtering to make their image forgeries appear more perceptually realistic.

In addition, the median filter's nonlinear properties make it useful for removing fingerprints left by other editing operations. It is has recently been incorporated into anti-forensic algorithms designed to hide traces of resampling [9] and evidence of compression [7]. Furthermore, median filtering may affect the effectiveness of different steganalysis techniques [1], [14].

While existing techniques have been developed to detect the use of median filtering [1]–[4], their performance is degraded in several important scenarios. This is particularly true when these detectors are held to low false positive rates. For example, the performance of existing median filtering detectors declines noticeably when testing on an image that has been JPEG compressed. This is problematic since many images are JPEG compressed during storage, capture, or transmission. Furthermore, the performance of these techniques degrades severely when small windows of an image are analyzed for evidence of localized median filtering. Additionally, existing techniques can encounter difficulties distinguishing median filtering from other editing operations at low false positive rates.

In this paper, we propose a new, robust median filtering forensic technique. It operates by analyzing the statistical properties of an image's median filter residual (MFR), which we define as the difference between an image in question and a median filtered version of itself [23]. This differs from existing techniques, which extract median filtering detection features directly from an image's pixel values or the pixel difference. By analyzing an image's MFR, we are able to suppress image content which may interfere with median filtering detection. To capture the statistical properties of the MFR, we fit it to an autoregressive (AR) model. We then train a support vector machine (SVM) to use the AR coefficients as features for median filter detection.

We test the effectiveness of our proposed median filter detection techniques through a series of experiments. Our experimental results show that the MFR can be used to detect me-

Manuscript received January 14, 2013; revised April 23, 2013 and June 26, 2013; accepted July 08, 2013. Date of publication July 15, 2013; date of current version August 15, 2013. This work was supported in part by NSFC (Grants 61070167, 61379870, U1135001), in part by the Research Fund for the Doctoral Program of Higher Education of China (Grant 20110171110042), and in part by NSF of Guangdong Province (Grant 2013020012788). The associate editor coordinating the review of this manuscript and approving it for publication was Prof. C.-C. Jay Kuo.

X. Kang was with the Department of Electrical and Computer Engineering, University of Maryland, College Park, MD 20742 USA. He is now with School of Information Science and Technology, Sun Yat-Sen University, Guangzhou, GD 510006, China (e-mail: isskxg@mail.sysu.edu.cn).

M. C. Stamm and K. J. R. Liu are with the Department of Electrical and Computer Engineering, University of Maryland, College Park, MD 20742 USA (e-mail: mcstamm@umd.edu; kjrliu@umd.edu).

A. Peng is with the School of Information Science and Technology, Sun Yat-Sen University, Guangzhou, GD 510006, China (e-mail: isskxg@mail.sysu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIFS.2013.2273394

dian filtering in JPEG compressed images with quality factors as low as 30 and in image windows as small as 32×32 pixels. It is capable of differentiating 3×3 median filtering from 5×5 median filtering. Additionally, our proposed method can distinguish between median filtering and other manipulations, such as Gaussian filtering, average filtering, and rescaling. Our experimental results demonstrate that our proposed method not only achieves better performance than existing median filtering detection techniques, but it does so using a substantially smaller feature set.

The rest of this paper is organized as follows. We review existing work on median filtering detection in Section II. In Section III, the median filter residual is introduced and our proposed detection technique is described. In Section IV, we evaluate the performance of our proposed algorithm and compare its performance with state-of-the-art techniques [1], [2], [4]. Finally, we conclude this paper in Section V.

II. Background and Prior Work

The median filter operates by replacing a pixel's value with the median value of the pixels in a small window surrounding it. The most commonly used median filter windows are squares of size 3×3 and 5×5 pixels. For the purposes of this work, we assume that median filtering is performed using a square $w \times w$ pixel window, where w is odd. Given an image, we can formally define the $w \times w$ median filter as shown in (1), at the bottom of the page, where $x(i,j)$ is the pixel value at point $(i,j), i,j \in Z$. A well known property of the median filter is that unlike linear filters, it is capable of smoothing an image while preserving its edges. As a result, the median filter is often used as a denoising filter.

Given a stochastic input to the median filter, the median filter's highly nonlinear nature makes it difficult to theoretically analyze the relationship between its input and output. Bovik was able to demonstrate that median filtering often produces constant or nearly constant regions called streaks within an image [10]. Bovik analyzed this phenomenon quantitatively and obtained the probability that the median values stemming from overlapping windows are equal [10].

Early forensic work capable of detecting median filtering made use of fingerprints left by a digital camera's color filter array (CFA) pattern and interpolation coefficients. Swaminathan *et al.* modeled tampering operations as linear filters, then estimated the tamper filter applied to an image using blind deconvolution with the CFA pattern and interpolation coefficients as constraints [11]. Chuang used a similarly constrained blind deconvolution algorithm to estimate the empirical frequency response of a tampering operation [12]. While these early techniques can successfully detect median filtering, they require either an accurate estimate or direct knowledge of the camera model used to capture an image. As a result, their performance is sensitive to the training data used.

Kirchner and Fridrich proposed a pair of median filter detectors inspired by the streaking artifacts discovered by Bovik [1]. To identify the presence of streaking artifacts in an image $x(i,j)$, Kirchner and Fridrich examined statistical properties of the image's first order pixel difference:

$$e_{i,j}^{(k,l)} = x(i,j) - x(i+k, j+l), \tag{2}$$

where

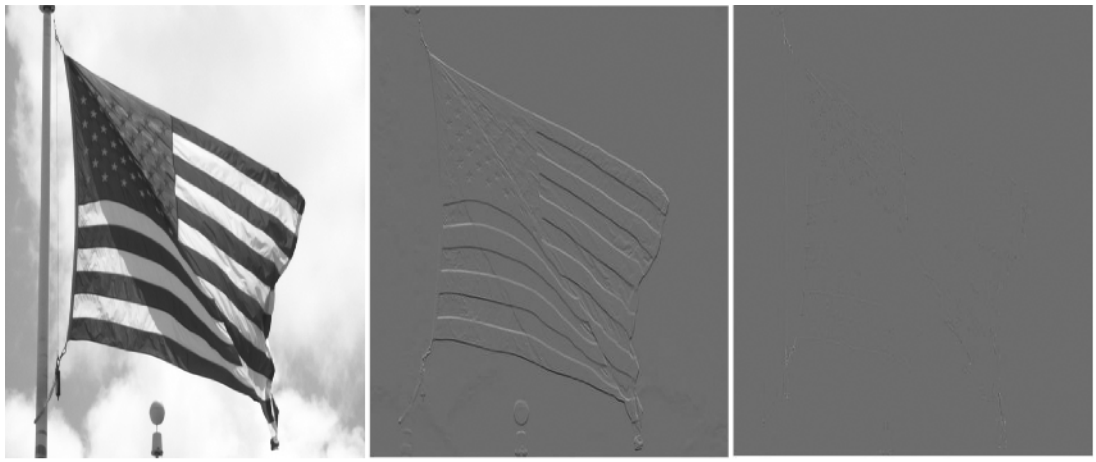
$$(k,l) \in \{(0,1),(0,-1),(1,0),(-1,0), (1,1),(1,-1),(-1,1),(-1,-1)\}$$

and $x(i,j)$ is the pixel value at point (i,j), $i, j \in Z$. Defining the histogram of $e_{i,j}^{(k,l)}$ values as $\{\ldots, h_{-1}^{(k,l)}, h_0^{(k,l)}, h_1^{(k,l)}, \ldots\}$, they proposed a simple median filtering detector that operates by comparing the ratio $\rho^{(k,l)} = h_0^{(k,l)}/h_1^{(k,l)}$ to a decision threshold. Additionally, they proposed a more robust detector using subtractive pixel adjacency matrix (SPAM) features. The set of SPAM features are the set of distributions of a first order pixel difference conditioned on each possible value of the neighbor first order difference [15]. Kirchner and Fridrich demonstrated that SPAM features can be used to detect median filtering in high to medium quality JPEG compressed images. The detector's performance degrades, however, as the JPEG's quality factor decreases. This is particularly true at low probabilities of false positive. Additionally, since a large number of observations are required to obtain good estimates of these conditional first order difference distributions, SPAM's performance degrades as the number of pixels in an image or image window decreases which was indicated in [2]. This is particularly important when performing localized median filtering detection through block-wise analysis.

Similarly, the authors of [3] proposed detecting median filtering by analyzing the probability that an image's first order pixel difference will be zero in textured regions. Furthermore, they demonstrated that their technique can distinguish median filtering from rescaling, Gaussian filtering, and average filtering. While they were able to demonstrate that this technique can very effectively detect median filtering in uncompressed images, its performance degrades significantly in JPEG compressed images.

The median pixel values obtained from overlapping filter windows related to one another since overlapping windows share several pixels in common. Yuan proposed detecting median filtering by measuring the relationships among pixels within a 3×3 window [2]. This is done by extracting a set of 44 features, known as the median filtering feature set (MFF),

$$\mathrm{med}_w(x(i,j)) = \mathrm{median}\left\{x(i+h, j+v) | h, v \in \left(-\frac{w-1}{2}, \ldots, 0, \ldots, \frac{w-1}{2}\right)\right\} \tag{1}$$



(a) (b) (c)

Fig. 1. Example showing (a) an image, (b) its first order difference, and (c) its median filter residual.

from an image. These sets include features such as the distribution of the block median pixel value and the distribution of the number distinct gray levels within a window. Yuan [2] demonstrated that the MFF method can achieve comparable or better performance than the SPAM on both high and moderate quality JPEGs and when detecting median filtering on small image windows. However, as with the Kirchner and Fridrich's technique, the performance of Yuan's technique decreases as the JPEG quality factor is lowered or as the image size examined shrinks.

The authors in [24] calculated the edge based prediction matrix (EBPM) of different kinds of image edges and obtained 72 dimensions of prediction coefficients to differentiate median filtering. They use a prediction model of the *pixel values in image regions with different gradients* and capture statistical relationships between nearby pixels to perform median filtering detection. In their recent work [4], they exploited cumulative distribution function of 1st-order and 2nd-order image difference as fingerprints to construct the global probability feature set (GPF). They also used the local correlations between different adjacent image difference pairs to construct the local correlation feature set (LCF). They finally used GPF and LCF to construct a new feature set GLF of 56 dimensions. Their method achieved good performance for low resolution and JPEG compression.

III. AR Model of Median Filter Residual

Existing median filtering detectors extract their detection features directly from the pixel values or the pixel difference of the image being examined. As a result, image content such as edge or texture information and the block artifacts from JPEG compression may interfere with attempts to capture statistical traces of median filtering. Take for example the first order pixel difference used by several detectors [1], [3], [4].Fig. 1 shows an image along with its first order pixel difference taken in the horizontal direction. We can clearly see in this figure that the first order difference contains a great deal of the image's edge content. This edge information and the block artifacts may affect the conditional first order difference distributions used by SPAM to detect median filtering. We note that while the MFF feature set does not include first order pixel differences, the MFF features are similarly affected by edge content.

To suppress both image content and block artifacts, and develop a more robust median filtering detection technique, we propose extracting detection features from the difference between a median filtered version of an image and the image itself. We refer to this difference as an image's median filter residual (MFR), which we formally define as

$$\begin{aligned} d(i,j) &= \mathrm{med}_w(y(i,j)) - y(i,j) \\ &= z(i,j) - y(i,j) \end{aligned} \tag{3}$$

where $y(i,j)$ is original pixel value at point (i,j) and $z(i,j)$ is median filtered value of $y(i,j)$. In this work, we use $w = 3$ when calculating an image's MFR. We can see from Fig. 1(c) that the median filter residual contains less edge information than the first order pixel difference.

To understand how the MFR can be used to detect median filtering in an image y, let us examine properties of the MFR when y is unaltered and when y has been median filtered. Median filtering detection can be framed as differentiating between the following two hypotheses:

H_0: y is not a median filtered image, i.e., $y = x$, where x is an unaltered image.

H_1: y is a median filtered image, *i.e.,* $y = \mathrm{med}_u(x)$.

We note that the median filter window size w used to obtain the MFR need not be the same as the median filter window size u used when altering the image.

Under hypothesis H_0, y is equal to an unaltered image x, therefore

$$d(i,j) = \mathrm{med}_w(x(i,j)) - x(i,j), \tag{4}$$

and

$$z(i,j) = \mathrm{med}_w(x(i,j)). \tag{5}$$

In this scenario, the value of $z(i,j)$ could potentially be equal to the value of $x(k,l)$ for any (k,l) that lies in the $w \times w$ median filter window surrounding (i,j). An example of this is shown

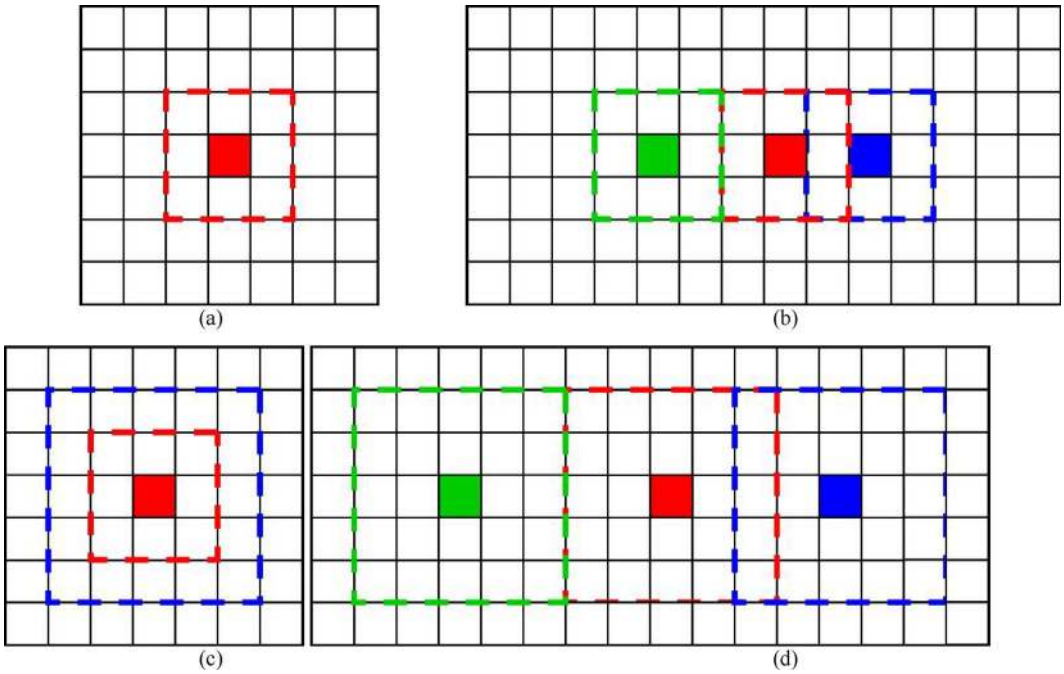


Fig. 2. Example showing (a) the $w \times w$ median filter window of the MFR of a pixel under hypothesis H_0, and (b) the overlap between of the median filter windows of the MFR (in red and blue) under hypothesis H_0 along with (c) the $u \times u$ modifying median filter window (in red) and the effective $(w+u-1-1) \times (w+u-1)$ median filter window of the MFR (in blue) of a pixel under hypothesis H_1, and (d) the overlap between the median filter windows of the MFR (in red and blue) under hypothesis H_1. In this example, $w = 3$ and $u = 3$.

in Fig. 2(a), where the pixel value $z(i,j)$ (shown in red) can be equal to the value of any $x(k,l)$ in the dashed red box. Because of this, any two distinct MFR values $d(i,j)$ and $d(i+h,j+v)$ could have z terms corresponding to the same x value as long as $h, v < w$. This is illustrated in Fig. 2(b), where two $w \times w$ windows with less than w pixels displacement will overlap.

Under hypothesis H_1, y is equal to a median filtered version of x, i.e.,

$$y(i,j) = \text{med}_u(x(i,j)). \tag{6}$$

As a result

$$d(i,j) = \text{med}_w(\text{med}_u(x(i,j))) - \text{med}_u(x(i,j)) \tag{7}$$

and

$$z(i,j) = \text{med}_w(\text{med}_u(x(i,j))) \tag{8}$$

The value of $z(i,j)$ can be equal to the value of $y(s,t)$ for any (s,t) that lies in the $w \times w$ median filter window surrounding (i,j). However, the value $y(s,t)$ can be equal to the value of $x(k,l)$ for any (k,l) that lies in the $u \times u$ median filter window surrounding (s,t). As a result, value of $z(i,j)$ can be equal to the value $x(k,l)$ for any (k,l) in the $(w+u-1) \times (w+u-1)$ window surrounding (i,j). An example of this is shown in Fig. 2(c). Because of this, under hypothesis H_1 any two distinct MFR values $d(i,j)$ and $d(i+h,j+v)$ could have z terms corresponding to the same x value as long as $h, v < w+u-1$. This phenomenon is shown in Fig. 2(d).

Let us refer to the window over which the z term of two different d values can correspond to the same x value as the shared value window. Examining the shared value window under each hypothesis, we can observe the following:

H_0: The MFR's shared value window is of size $w \times w$.

H_1: The MFR's shared value window is of size $(w+u-1) \times (w+u-1)$.

Because the size of the shared value window changes under each hypothesis, the relationship between $d(i,j)$ and its neighbors will also change under each hypothesis.

To capture this effect using a feature set of low dimensionality, we fit the MFR to an autoregressive (AR) model. Because an AR model essentially performs linear prediction, the values of the AR coefficients depend heavily on how the MFR values of nearby pixels relate to one another. Since the shared value window of the MFR is smaller under hypothesis H_0 than under H_1, the coefficients of the AR model will be substantially different if the image in question has been median filtered. As a result, we use the AR coefficients of the MFR as features when performing median filtering detection.

To further reduce the dimensionality of our model, we assume that an image's statistical property is the same in the horizontal and vertical directions. Using this assumption along with the fact that median filter windows are symmetric, we fit the MFR to a one dimensional AR model in the row direction

$$d(i,j) = -\sum_{k=1}^{p} a_k^{(r)} d(i,j-k) + \varepsilon^{(r)}(i,j), \tag{9}$$

and in the column direction

$$d(i,j) = -\sum_{k=1}^{p} a_k^{(c)} d(i-k,j) + \varepsilon^{(c)}(i,j), \tag{10}$$

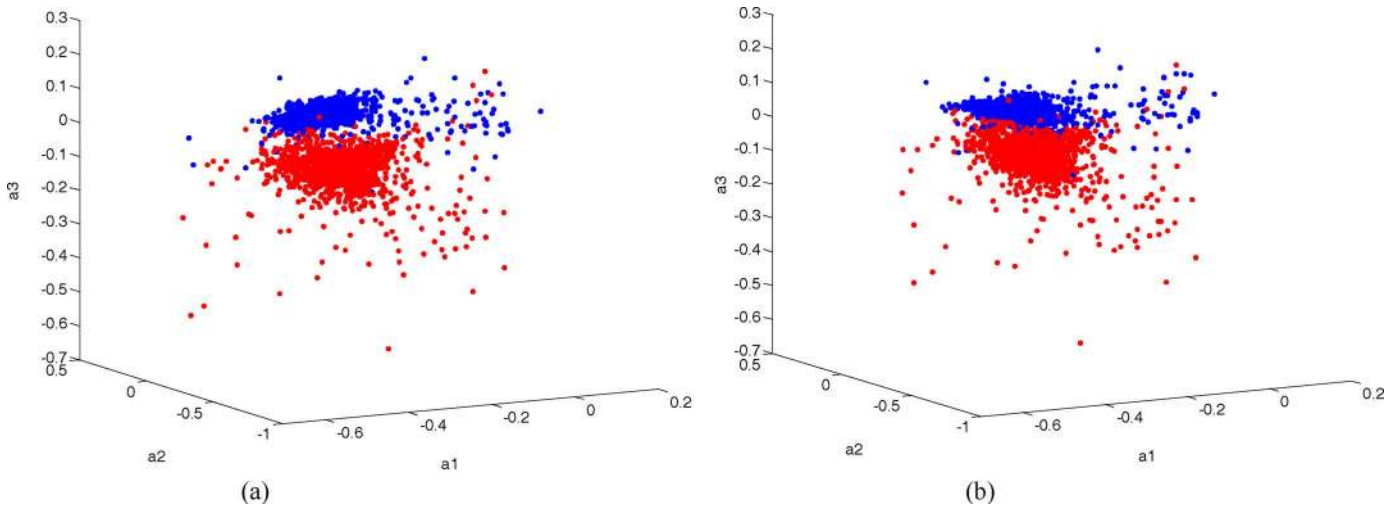


Fig. 3. Plot of the first three AR coefficients of the MFR for (a) unaltered images (red) and the 3×3 median filtered images (blue); (b) JPEG 70 compressed images (red) and the 3×3 median filtered then JPEG 70 compressed image (blue) in UCID database.

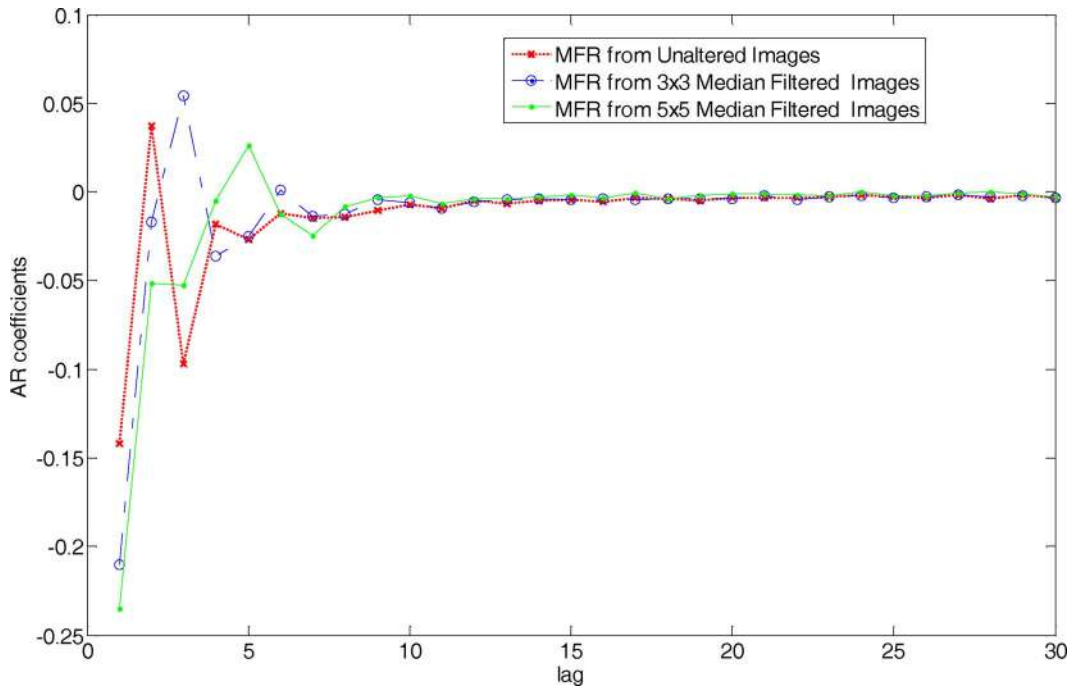


Fig. 4. Average AR coefficients of the MFR from unaltered images (red), the 3×3 median filtered images (blue), and the 5×5 median filtered images (green), respectively.

where $\varepsilon^{(r)}(i,j)$ and $\varepsilon^{(c)}(i,j)$ are the prediction errors [20] in the row direction and column direction respectively, p refers to order of AR model and $a_k^{(r)}$ and $a_k^{(c)}$ are the AR coefficients in the row direction and column direction respectively. We then average the AR coefficients in both directions to obtain a single, one dimensional AR model.

Fig. 3(a) shows the first three AR coefficients (a_1, a_2, a_3) of the MFR extracted from both unaltered and median filtered versions of images in the Uncompressed Color Image Database (UCID) [18]. Fig. 3(b) shows the first three AR coefficients of the MFR extracted from the same images after they have undergone JPEG compression with quality factor of 70. From these figures, we can clearly see that the unaltered and median filtered images can be separated on the basis of the MFR's AR coefficients. Furthermore, these figures show that JPEG compression has little effect of the ability to separate median filtered from unaltered images on the basis of their MFR's AR coefficients. This demonstrates the robustness of the MFR's AR coefficients to JPEG compression.

Fig. 4 shows the average value of the first 30 AR coefficients of each image in the UCID. From this figure, we can see that the AR coefficient values differ on average for roughly the first 10 AR coefficients. After this point, the AR coefficients are approximately the same regardless of whether or not an image was median filtered. Additionally, this figure shows that the largest AR coefficient occurs at different k's depending on whether or not an image was median filtered. This reinforces the notion that the AR coefficients are good features for median filtering detection.

To identify median filtering, we use a support vector machine trained on the first 10 AR coefficients of the MFR. While we have experimentally found that using 10 AR coefficients results in a desirable tradeoff between detection accuracy and the dimensionality of the feature space in the detection of both 3×3 and 5×5 median filtering, we have observed that 3×3 median filtering detection can still be accurately performed using as few as 4 AR coefficients. We note that the SPAM features proposed by Kirchner and Fridrich are 686 dimensions [1], the GLF [4] features are 56 dimensions, and the MFF features proposed by Yuan [2] are 44 dimensions. Since our method uses only 10 features, we are able to achieve a 1 to 2 order of magnitude reduction in the dimensions of the feature vector.

Our complete median filtering detection technique can be summarized as follows

1. Calculate an image's MFR using (3).
2. Fit the MFR to an AR model of order 10 in the row direction and in the column direction using all MFR values.
3. Average corresponding AR coefficients across each model acquired in Step 2 to obtain a single AR model.
4. Input the AR coefficients to an SVM trained to classify between median filtered and unaltered images.

IV. Experimental Results

To evaluate the effectiveness of our proposed median filtering detector and to compare its performance to existing median filtering detection techniques, we tested our proposed technique along with several others on UCID [18] and a composite image database which contains 6690 different kinds (such as raw images, rescanned images and rescaled images) of images from UCID, the BOSS RAW database (BR) [25], the BOWS2 image database (BOWS2) [26], the Dresden Image Database (DID) [27] and the NRCS Photo Gallery (NRCS) [28]. Each database (UCID, BR, BOWS2, DID, NRCS) contributes 1338 images with size of 512×384 to compose the composite image database. These databases are widely used to evaluate the performance of forensic techniques [2]–[7], and they are described in detail in [2] and [4]. The UCID database consists of 1338 uncompressed RGB images of size 512×384. The images in the other four databases (BR, BOWS2, DID, NRCS) are cropped to the size of 512×384 from the center of its full size source images. Then all color images were first converted to gray scale images before further processing. Median filtered images were generated by performing 3×3 median filtering and 5×5 median filtering on the unaltered gray-scale images. Each unaltered and median filtered image was then saved in both its uncompressed state and JPEG compressed state using a variety of quality factors ranging between 90 and 30.

We compared our proposed AR method with the SPAM method [1], the MFF method [2], and the GLF method [4]. We performed SVM training and testing for each of the four methods in the same manner. To perform classification, we used a C-SVM with a Gaussian kernel [19]

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$$

During cross-validation, once a training set was selected, we found the best kernel parameters for the SVM by performing an additional five-fold cross-validation in conjunction with a grid search. The grid search for the best parameters was performed on the multiplicative grid $(C, \gamma) \in \{(2^i, 2^j) | 4 \times i, 4 \times j \in Z\}$. Once the best parameters were identified, we used those parameters to get the classifier model on the entire training set. We then use the trained classifier model to perform a classification on the testing set.

Experimental results were reported on the UCID database in items *A)-D)*, and on the composite database in items *E)*. Four-fold cross validation was used to evaluate the effectiveness of each approach when testing on the UCID database. Specifically, the images in the UCID database were randomly divided into four folds of nearly equal size. In each repetition, the training set was composed of three folds (about 1003 images), while the remaining fold was used as the testing set (about 335 images). After four-fold cross-validation testing, we can obtain the detection results and ROC curve of all 1338 images in UCID database.

In real world scenarios, an investigator must often perform detection with a low probability of false positives. Because of this, each detector's performance at low false positive rate is critical. To take this into account, we report the performance of each detection technique at a low false positive rate such as 1%. Additionally, we report the minimal average decision error P_e of each technique under the assumption of equal priors and equal costs,

$$P_e = \min\left(\frac{P_{fp} + 1 - P_{tp}}{2}\right) \tag{11}$$

where P_{fp} and P_{tp} denote the false positive (FP) and true positive rates (TP), respectively.

A. Detecting Globally Applied Median Filtering

To measure the performance of our proposed method under ideal conditions, we performed median filtering detection on the set of uncompressed images. The results of this experiment are shown in Fig. 5(a) and (b)which show ROC curves obtained for each detection technique when tested against images modified using 3×3 and 5×5 median filters respectively. In Fig. 5(a), "Original VS MF3" denotes that the original unmodified image set versus the 3×3 median filtered image set. From these results, we can see that all four methods have comparable performance and achieve perfect or nearly perfect detection.

Next, we tested each technique's ability to detect 3×3 median filtering in images that were JPEG compressed using quality factors ranging between 90 and 30. ROC curves obtained from these experiments are shown in Fig. 6 and significant results are listed in Table I."MF3+JPEG70" denotes the composite operation of median filtering followed by JPEG compression with quality factor (QF) 70. For each JPEG quality factor test, our detector achieved a lower P_e than all other three methods. Additionally, the ROC curves show that our detector achieved a higher P_{tp} than all other detectors at all P_{fp} Rates. This is especially true at low false positive rates. At $P_{fp} = 1\%$, our detector achieved a $P_{tp} = 97.5\%$ when testing on images compressed using a quality factor of 70, while the MFF detector achieved a $P_{tp} = 56.1\%$, and the GLF detector

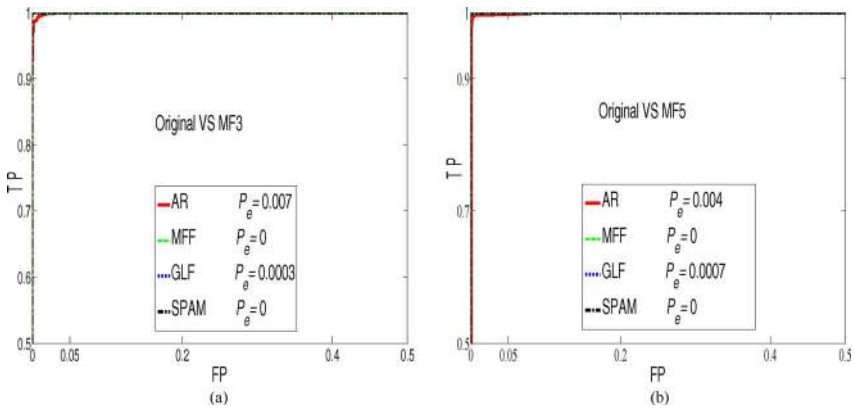


Fig. 5. ROC curves showing 3 × 3 median filtering (a) and 5 × 5 median filtering (b) detection performance on uncompressed images.

TABLE I
P_e AND P_{tp} AT $P_{fp} = 1\%$ FOR MEDIAN FILTERING DETECTORS AGAINST JPEG COMPRESSION. (THE BEST RESULT FOR EACH TRAINING–TESTING PAIR IS DISPLAYED WITH BOLD TEXTS.)

		MF3				MF5			
		AR	MFF	GLF	SPAM	AR	MFF	GLF	SPAM
JPEG 30	P_{tp}(%)	**55. 8**	21. 8	21. 7	4. 3	**76. 9**	32. 8	52. 5	56. 9
	P_e(%)	**2. 8**	10. 0	7. 0	21. 3	**2. 5**	5. 2	3. 4	6. 3
JPEG 50	P_{tp}(%)	**93. 5**	38. 5	46. 4	6. 6	**95. 8**	62. 4	85. 0	72. 9
	P_e(%)	**2. 2**	7. 2	5. 0	16. 1	**2. 1**	4. 2	2. 7	4. 0
JPEG 70	P_{tp}(%)	**97. 5**	56. 1	75. 5	26. 9	**98. 8**	70. 8	95. 5	93. 5
	P_e(%)	**1. 3**	4. 6	4. 3	10. 7	**1. 0**	3. 5	1. 8	2. 8
JPEG 90	P_{tp}(%)	**99. 5**	95. 3	97. 2	92. 8	**99. 7**	96. 3	99. 5	98. 8
	P_e(%)	**0. 5**	2. 2	1. 5	2. 5	**0. 5**	2. 0	0. 7	1. 0

achieved a $P_{tp} = 75.5\%$ and the SPAM detector achieved a $P_{tp} = 26.9\%$. This corresponds to P_{tp} improvements of 41.4%, 22.0% and 70.6% respectively. Similarly for images compressed using a quality factor of 50, our detector achieved a $P_{tp} = 93.5\%$ at $P_{fp} = 1.0\%$, while the MFF, GLF and SPAM detectors achieved $P_{tp} = 38.5\%$, $P_{tp} = 46.4\%$, and $P_{tp} = 6.6\%$ respectively. This corresponds to P_{tp} improvements of 55%, 47.1%, and 86.9% respectively. These results demonstrate that our proposed detection method is more robust to JPEG compression than existing techniques. It can also be observed from Table I and Fig. 6 that our detection method's advantage over the other three methods increases as the JPEG quality factor decreases.

A similar improvement in performance over the state-of-the-art MFF and GLF methods were observed when we repeated the experiment using 5 × 5 median filtering. Detailed results of this experiment are shown in Table I. From Table I, we can see that our proposed method achieved a larger $P_{tp} = 76.9\%$ at $P_{fp} = 1.0\%$ than the other three methods for images compressed with a quality factor of 30. These experimental results show that the performance of the AR classifier remains strong when the JPEG compression quality factor is as low as 30 in detection of 5 × 5 median filtering.

B. Detecting Median Filtering in Low-Resolution Images and Image Windows

The ability to detect median filtering in low-resolution images and image windows is essential for detecting forgeries when a portion of a median filtered image is inserted into a nonmedian filtered image. To test each detector's performance on small image windows, we created a database to test image blocks by cropping a block of size 128 × 128, 64 × 64 and 32 × 32 from the center of an image. The state of the art median filtering detectors when operating on small image windows are the MFF method [2] and the GLF method [4]. For the sake of brevity, we only compared our method with both the MFF method and the GLF method on JPEG 70 compressed images. ROC curves obtained from this experiment are shown in Fig. 7.

From Fig. 7, we can see that the performance of our proposed AR detector is stronger than that of the MFF and GLF detectors for blocks with sizes as low as 32 × 32. Our AR method achieved a $P_{tp} = 69.7\%$ at $P_{fp} = 1\%$ when testing on 128 × 128 pixel blocks compressed with a quality factor of 70, while the GLF and MFF methods achieved a P_{tp} of 28.6% and 9.8% respectively at $P_{fp} = 1\%$. This corresponds to P_{tp} improvements of 41.1% and 59.9% compared with GLF

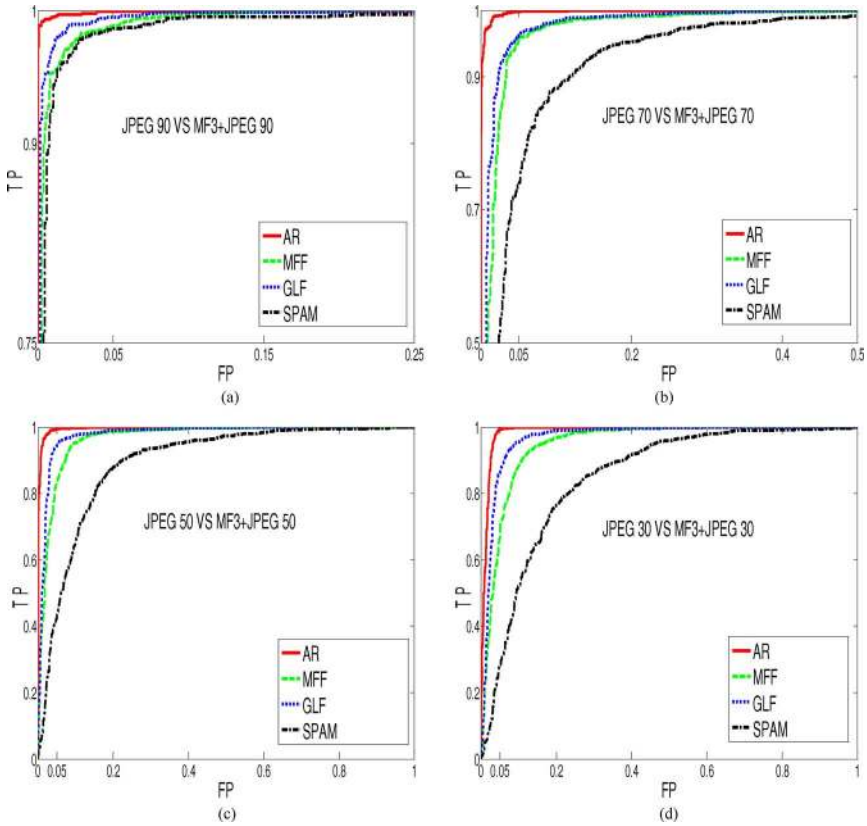


Fig. 6. ROC curves showing 3×3 median filtering detection performance on (a) JPEG 90 compressed images, (b) JPEG 70 compressed images, (c) JPEG 50 compressed images, and (d) JPEG 30 compressed images. Different scales were applied on x axis and y axis for clear demonstration.

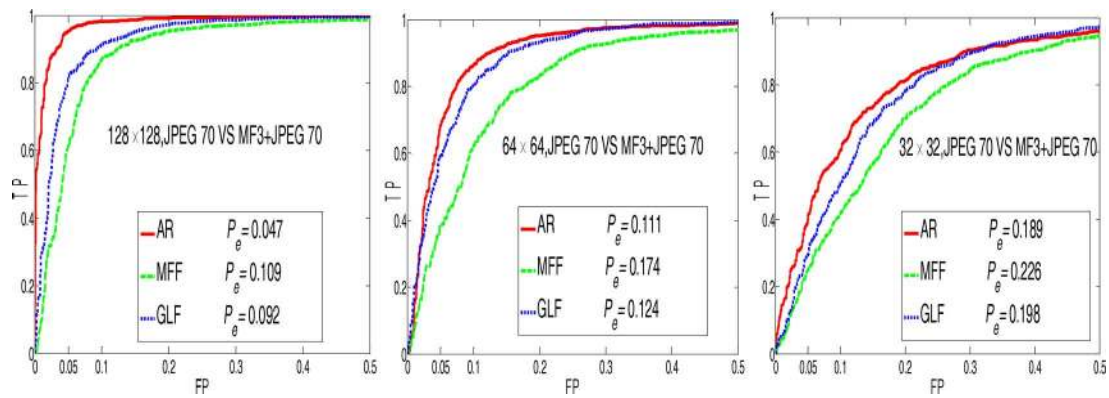


Fig. 7. ROC curves showing 3×3 median filtering detection performance on JPEG compressed images of size 128×128 (left), 64×64 (middle), 32×32 (right).

and MFF. For blocks of size 64×64, our detector achieved a $P_{tp} = 68.2\%$ at $P_{fp} = 5\%$. For 32×32 pixel blocks, our detector achieved a $P_{tp} = 60.5\%$ at $P_{fp} = 10\%$. We obtained similar results when testing on blocks from images modified by 5×5 median filtering.

An example of a cut-and-paste image forgery and corresponding forensic detection results were shown in Fig. 8. Fig. 8(a) shows the 3×3 median filtered image from which an object (the woman on the left) was cut. Fig. 8(b) shows the unaltered image into which the cut object was pasted. Fig. 8(c) shows the composite image, which had been JPEG compressed using a quality factor of 70. In order to detect the forgery, the composite image was first segmented into 128×128 pixel blocks, then each block was tested for evidence of locally applied median filtering. In this example, each detection method tested was trained on 128×128 pixel blocks from images in UCID database that had been compressed using a quality factor of 70. Blocks corresponding to median filtering detections are boxed and outlined in red. Fig. 8(d) shows the result of blockwise detections on the composite image using our proposed AR method. In this example, each of the outlined blocks contains pixels corresponding to the inauthentic object and the pasted object can be detected correctly using our proposed AR method. Fig. 8(e) shows the result of blockwise detections on the composite image using the GLF method. In this example, multiple false alarms occur and the detection rate

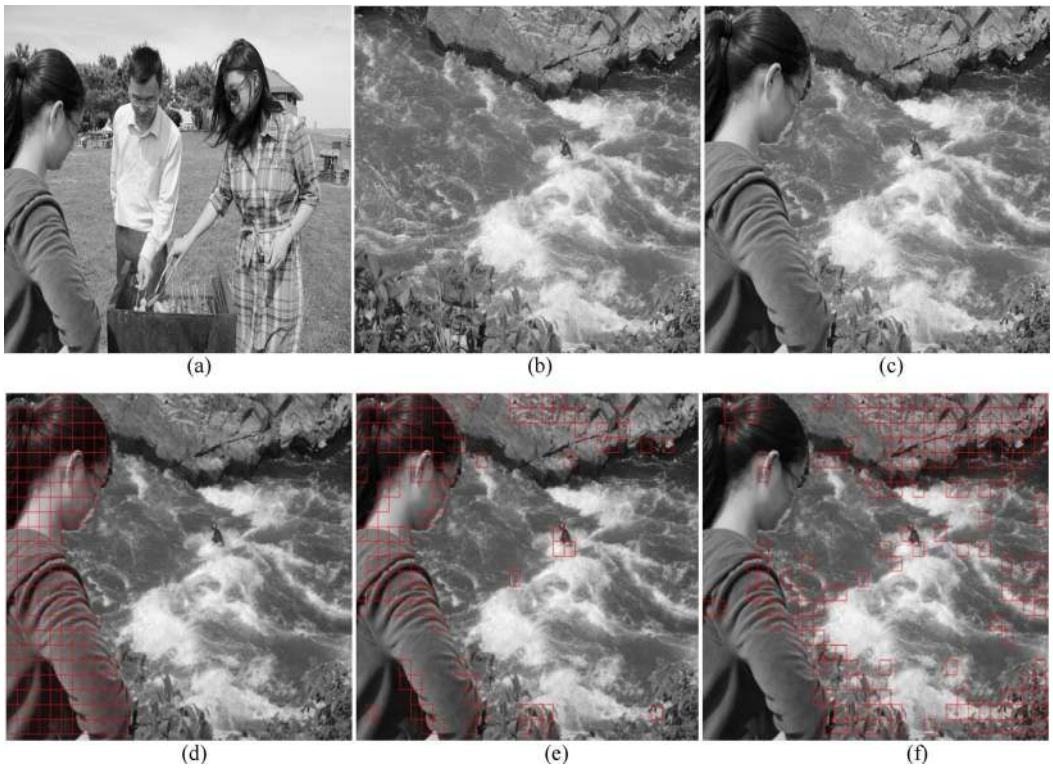


Fig. 8. Cut and paste forgery detection example showing (a) the median filtered image from which an object is cut, (b) the unaltered image into which the cut object is pasted, (c) the composite image which is JPEG compressed using a quality factor of 70. Blocks detected as median filtered are outlined in red boxes: (d) blockwise detections using the AR method, (e) blockwise detections using the GLF method, and (f) blockwise detections using the MFF method.

is decreased. Fig. 8(f) shows the result of blockwise detections on the composite image using the MFF method. In this case, the inauthentic object cannot be located with the MFF method correctly. This example shows that our method achieves the best performance in the cut-and-paste forgery detection.

C. Distinguishing Median Filtering From Other Manipulations

Identifying the particular operation used to alter an image is an important forensic problem. This can be difficult in the case of median filtering, because several other operations such as linear smoothing and resizing leave behind similar forensic traces.

We tested the ability of our proposed AR method, along with the SPAM, GLF and MFF methods, to differentiate between median filtering and other popular tools, including 3×3 Gaussian filtering with $\sigma = 0.5$ (GAU), 3×3 average filtering (AVE), upscaling (UpRes) and downscaling (DownRes). Bilinear interpolation was used to perform both upscaling and downscaling. The upscaling factor was set to 1.1, while the downscaling factor was 0.9.

To achieve a baseline measure of the performance of each technique, we first evaluated their ability to distinguish median filtering from other operations in uncompressed images. Our experimental results show that under these ideal conditions, each technique was able to distinguish median filtering from other operations perfectly (i.e., each technique achieved a $P_e = 0\%$).

Next, we evaluated the performance of each technique on images that had been JPEG compressed using a quality factor of 70. This experiment reflects conditions more likely to be encountered by a forensic examiner in a real world scenario. ROC curves displaying the performance of method are shown in Fig. 9 for 3×3 median filtering. Additionally, detection results showing the P_e and P_{tp} at $P_{fp} = 1\%$ are displayed in Table II for both 3×3 and 5×5 median filtering.

These experimental results show that our method can discriminate between median filtering and other operations with high accuracy. As can be seen in Table II, the worst P_e value achieved by our detector among the four manipulations was only 2.3%. Furthermore, these results show that our method can achieve substantial performance gains over the other techniques at low false positive rates. For example, when testing against images which had been modified by Gaussian blurring and downscaling, our method achieved a $P_{tp} = 96.5\%$ and $P_{tp} = 95.6\%$ respectively at $P_{fp} = 1\%$ for 3×3 median filtering. At the same false positive rate, the best results of other three methods were achieved by GLF and its $P_{tp} = 76.4\%$, 69.3% respectively.

In practical settings, it is likely that an investigator will need to distinguish between median filtering and a collection of other operations rather than a single, known operation. To evaluate our proposed forensic technique's ability to do this, we pooled all of the images used in the previous experiments that were JPEG compressed with a quality factor of 50 into two different classes. Class1 contained the 13383 $\times$ 3 median filtered images, while Class2 was made up of 1338 images randomly chosen from the sets of unaltered images and images modified by average filtering, Gaussian filtering, upscaling, and downscaling. We then used the proposed AR method along with MFF, GLF

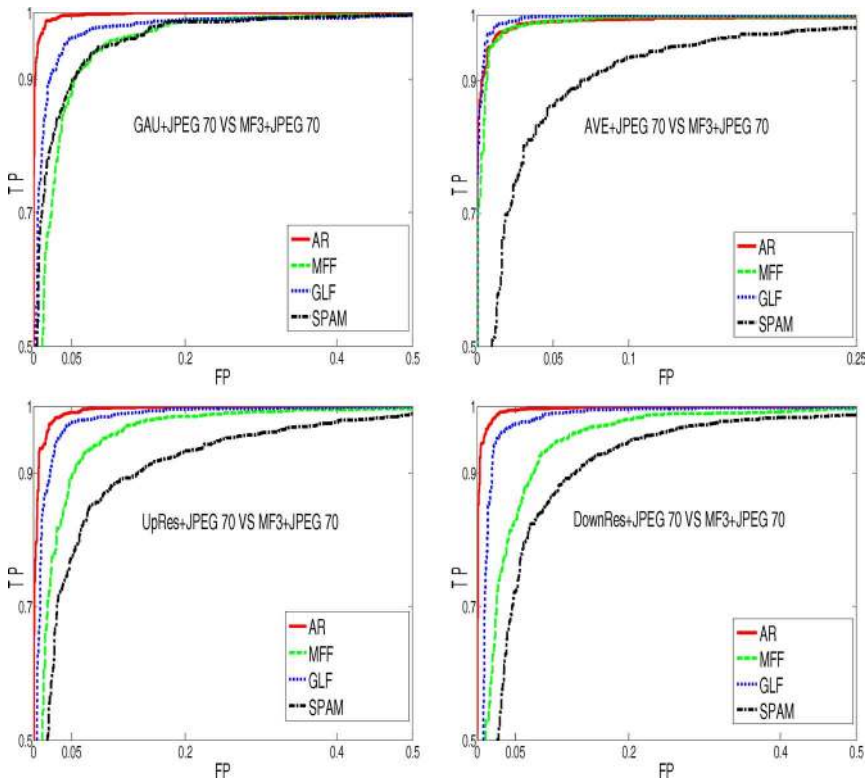


Fig. 9. ROC curves showing each technique's ability to discriminate 3×3 median filtering from Gaussian filtering (top left), average filtering (top right), upscaling (bottom left), and downscaling (bottom right) in JPEG compressed images using a quality factor of 70. Different scales were applied on x axis and y axis for clear demonstration.

TABLE II
P_e AND P_{tp} AT $P_{fp} = 1\%$ OF DISTINGUISHING MEDIAN FILTERING FROM OTHER MANIPULATIONS. (THE BEST RESULT FOR EACH TRAINING–TESTING PAIR IS DISPLAYED WITH BOLD TEXTS.)

		MF3				MF5			
		AR	MFF	GLF	SPAM	AR	MFF	GLF	SPAM
MF VS GAU	P_{tp}(%)	**96. 5**	44. 3	76. 4	68. 7	**99. 1**	66. 9	96. 7	96. 3
	P_e(%)	**1. 5**	7. 0	4. 4	6. 9	**0. 8**	4. 8	1. 7	1. 8
MF VS UpRes	P_{tp}(%)	**93. 7**	48. 2	76. 8	29. 6	**98. 9**	44. 3	94. 8	90. 4
	P_e(%)	**2. 3**	6. 8	3. 6	11. 1	**1. 0**	4. 8	1. 6	2. 3
MF VS DownRes	P_{tp}(%)	**95. 6**	43. 9	69. 3	23. 1	**99. 3**	75. 0	95. 8	94. 6
	P_e(%)	**1. 9**	7. 7	3. 6	11. 3	**0. 8**	4. 7	1. 5	2. 4
MF VS AVE	P_{tp}(%)	95. 2	95. 6	**96. 6**	51. 0	**99. 7**	97. 1	98. 0	93. 5
	P_e(%)	2. 1	2.1	**1. 4**	8. 2	**0. 5**	1. 7	1. 1	2. 4

and SPAM methods to distinguish between the two classes. The four-fold cross validation method was also used in this experiment.

ROC curves displaying the experimental performance of each technique with different image sizes are shown in Fig. 10. In Fig. 10, "ALL VS MF3 + JPEG 70" denoted that the images in both Class 2 and Class 1 were JPEG compressed with a quality factor 70. These results show that our proposed AR method can distinguish between median filtering and other operations better than other three techniques, especially on small sized images. On image sizes of 128×128 at a false positive rate of $P_{fp} = 5\%$, our proposed technique achieved a $P_{tp} = 90.1\%$. By contrast, the SPAM, the GLF and MFF techniques achieved $P_{tp} = 25.6\%$, $P_{tp} = 72.1\%$ and 51.1% respectively. This corresponds to improvements of 64.5%, 18.0% and 39.0% respectively.

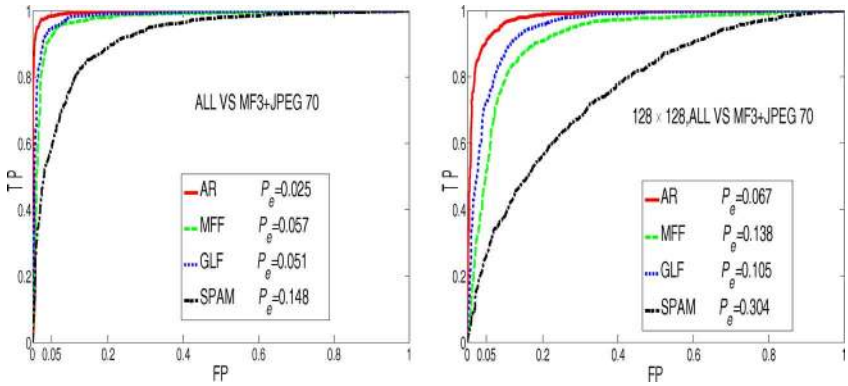


Fig. 10. ROC curves showing each technique's ability to discriminate median filtered images from nonmedian filtered images with size of 512 × 384 (left) and 128 × 128 (right).

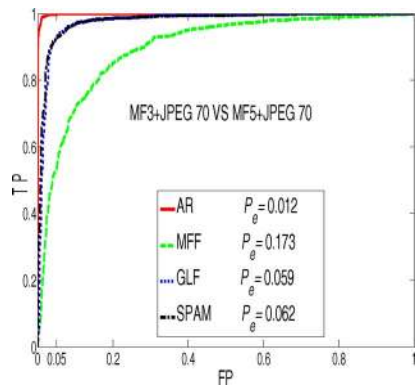


Fig. 11. ROC curves showing each technique's ability to discriminate 3 × 3 median filtering (MF3) from 5 × 5 median filtering (MF5) on JPEG compressed images using a quality factor of 70.

D. Differentiating 3 × 3 Median Filtering From 5 × 5 Median Filtering

Once a forensic investigator has identified that an image has been median filtered, they may wish to determine the window size used during median filtering. We have found that our AR method can differentiate between 3 × 3 and 5 × 5 median filtering with high accuracy. To experimentally verify this, we created 1338 3 × 3 and 1338 5 × 5 median filtered versions of each image in the UCID, and then JPEG compressed them with a quality factor of 70. Next, we used our proposed method along with the MFF, the GLF and SPAM techniques to distinguish between 3 × 3 and 5 × 5 median filtering. ROC curves showing the results of this experiment are displayed in Fig. 11. From these results, we find that at a false positive rate of $P_{fp} = 1\%$, our AR method achieved a much higher $P_{tp} = 98.2\%$ than other three methods. The MFF, GLF and SPAM techniques achieved $P_{tp} = 14.3\%$, $P_{tp} = 44.1\%$ and $P_{tp} = 62.2\%$ respectively. This corresponds to improvements of 83.9%, 54.1% and 36.0% respectively. These results show that our proposed technique can identify the median filter's window size more accurately than existing techniques.

E. Detection Results on a Composite Database

In addition, we evaluated the performance of our detector on the previously mentioned composite database consisting of 6690 images of size of 512 × 384 pixels. When testing on this database, the training set was chosen to contain 2676 images (40% of the database size) while the testing set contained the remaining 4014 images. Because the training and testing sets were sufficiently large, four-fold cross validation is not applied on the composite database. The setup is similar to Items *A)-D)*.

First, we evaluated each technique's ability to detect 3 × 3 median filtering in images that were JPEG compressed using quality factor 70. The results of this experiment are shown in Fig. 12(a). From this figure, we can see that our AR method outperforms all other three methods. At a false positive rate of $P_{fp} = 5\%$, the AR, GLF, SPAM and MFF methods achieved true positive rates of $P_{tp} = 83.8\%$, $P_{tp} = 76.7\%$, $P_{tp} = 67.2\%$ and $P_{tp} = 51.6\%$ respectively. Next, we repeated this experiment on images sized 128 × 128 pixels. These small images were cropped from the center of each full sized image in the composite database. The results of this experiment are shown in Fig. 12(b). These results demonstrate that our method is able to outperform all other techniques on small images and image windows. At $P_{fp} = 5\%$, our method achieved a $P_{tp} = 60.6\%$. This corresponds to a P_{tp} improvement of 10.5% over the second best performing GLF method.

All previous experiments using JPEG compression applied JPEG postcompression, that is, JPEG compression performed after median filtering. As JPEG compression is a popular image format, we tested whether JPEG compression before median filtering affected the performance of each detection technique. In this experiment, images in the Class 1 were first JPEG compressed using a quality factor of 90, then 3 × 3 median filtered, and finally saved in JPEG format with a quality factor of 70. Images in the Class 2 were JPEG compressed using a quality factor of 70. We then used each technique to perform median filtering detection and used the results to obtain the ROC curves shown in Fig. 12(c). In Fig. 12(c), Class 1 and Class 2 are denoted as "JPEG +MF3+JPEG 70" and "JPEG" respectively. From these results, we can observe that our proposed method is more robust against JPEG precompression and achieves best performance among all four methods. JPEG precompression has little effect on our proposed method in differentiating the two classes when comparing Fig. 12(a) with Fig. 12(c).

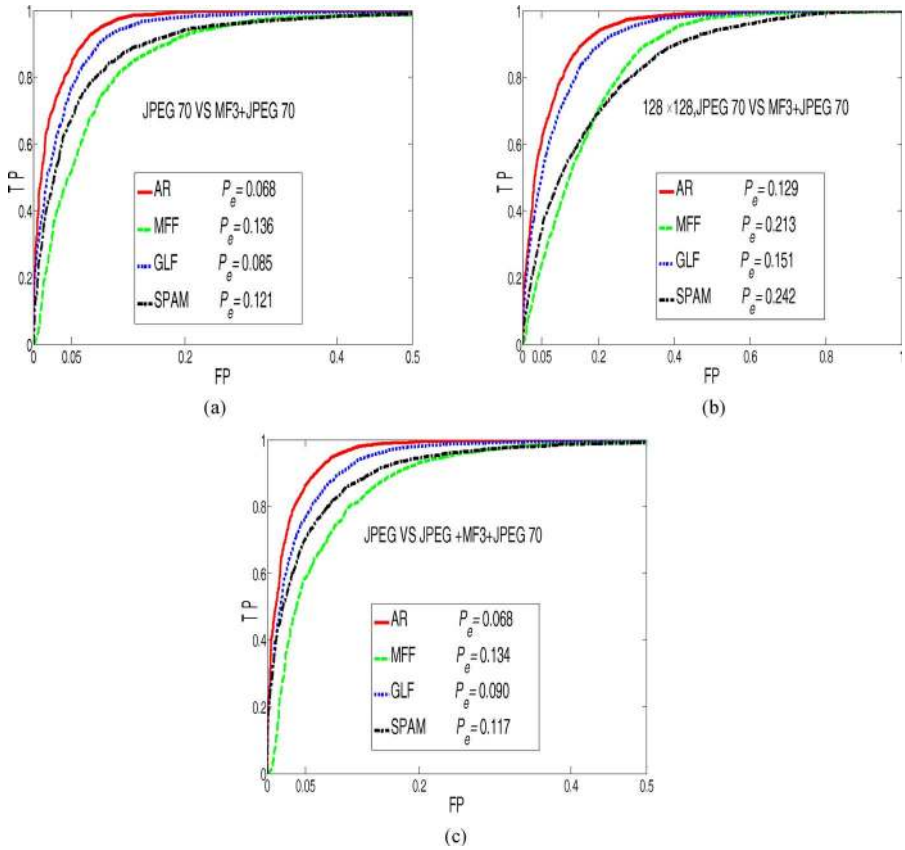


Fig. 12. ROC curves show each technique's performance on the composite database for 3×3 median filtering detection on (a) images of size 512×384 and (b) images of size 128×128. Plots in (c) demonstrates each method's ability to detect 3×3 median filtering when images of size 512×384 were preprocessed by JPEG compression with $\mathrm{QF} = 90$.

V. Conclusion

In this paper, we have proposed a new, robust median filtering detection technique. To reduce interference from an image's edge content and the block artifacts from JPEG compression, we proposed gathering detection features from an image's median filter residual. Specifically, we built a one dimensional AR model of an image's MFR and used the AR coefficients as median filtering detection features. Our AR features achieved a one to two order of magnitude reduction in the dimensionality of the detection feature space used by existing techniques such as the SPAM and MFF methods. We then used these features to train a support vector machine to perform median filtering detection.

Through a series of experiments, we have demonstrated that our proposed median filtering forensic technique outperforms existing detectors under a variety of scenarios. Our experimental results have shown that our technique can detect median filtering in images that have been JPEG compressed using quality factors as low as 30. We have demonstrated that our technique can identify median filtering in small image blocks. Using these results, we have shown that our proposed detector can be used to identify cut-and-paste forgeries. Additionally, our experimental results show that our proposed technique can more reliably distinguish between median filtering and rescaling editing operations than existing median filtering forensic techniques.

Our experimental results have shown that our detector achieves substantial performance gains over existing forensic techniques when the false positive rate is held low (e.g., $P_{fp} = 1\%$). Because median filtering detection must often be performed at low false positive rates, these results demonstrate that our proposed technique is better suited for use in real world scenarios than existing techniques.

Acknowledgment

The authors thank H. Yuan for providing the code of MFF scheme.

References

[1] M. Kirchner and J. Fridrich, "On detection of median filtering in digital images," in *Proc. SPIE, Electron. Imaging, Media Forensics and Security II*, 2010, vol. 7541, pp. 1–12.
[2] H. Yuan, "Blind forensics of median filtering in digital images," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 4, pp. 1335–1345, Dec. 2011.
[3] G. Cao, Y. Zhao, R. Ni, L. Yu, and H. Tian, "Forensic detection of median filtering in digital images," in *Proc. 2010 IEEE Int. Conf. Multimedia and EXPO 2010*, 2010, pp. 89–94.
[4] C. Chen, J. Ni, R. Huang, and J. Huang, "Blind median filtering detection using statistics in difference domain," in *Proc. Inform. Hiding 2012*, Berkerly, CA, USA, May 2012.
[5] A. C. Popescu and H. Farid, "Exposing digital forgeries by detecting traces of resampling," *IEEE Trans. Signal Process.*, vol. 53, no. 2, pt. 2, pp. 758–767, Feb. 2005.
[6] M. C. Stamm and K. J. R. Liu, "Forensic detection of image manipulation using statistical intrinsic fingerprints," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 3, pp. 492–506, Sep. 2010.
[7] M. C. Stamm and K. J. R. Liu, "Anti-forensics of digital image compression," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 3, pp. 1050–1065, Sep. 2011.

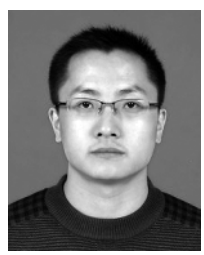
[8] A. E. Dirik and N. Memon, "Image tamper detection based on demosaicing artifacts," in *Proc. Int. Conf. Image Processing*, Cairo, 2009.
[9] M. Kirchner and R. Böhme, "Hiding traces of resampling in digital images," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 4, pp. 582–592, Dec. 2008.
[10] A. C. Bovik, "Streaking in median filtered images," *IEEE Trans. on Acoust., Speech and Signal Processing*, vol. 35, no. 4, pp. 493–503, Apr. 1987.
[11] A. Swaminathan, M. Wu, and K. J. R. Liu, "Digital image forensics via intrinsic fingerprints," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 1, pp. 101–117, Mar. 2008.
[12] W. H. Chuang, A. Swaminathan, and M. Wu, "Tampering identification using empirical frequency response," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Processing*, 2009, pp. 1517–1520.
[13] T. Pevný, P. Bas, and J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 2, pp. 215–224, Jun. 2010.
[14] A. D. Ker and R. Böhme, "Revisiting weighted stego-image stegoanalysis," in *Proc. SPIE, Electron. Imaging: Security, Forensics, Steganography and Watermarking of Multimedia Contents X*, 2008, vol. 6819, p. 5.
[15] T. Pevny and J. Fridrich, "Detection of double-compression in JPEG images for applications in steganography," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 2, pp. 247–258, Jun. 2008.
[16] W. Luo, J. Huang, and G. Qiu, "JPEG error analysis and its applications to digital image forensics," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 3, pp. 480–491, Sep. 2010.
[17] W. S. Lin, S. K. Tjoa, H. V. Zhao, and K. J. R. Liu, "Digital image source coder forensics via intrinsic fingerprints," *IEEE Trans. Inf. Forensics Security*, vol. 4, no. 3, pp. 492–506, Sep. 2009.
[18] G. Schaefer and M. Stich, "UCID-An uncompressed color image database," in *Proc. SPIE, Storage and Retrieval Methods and Applicat. for Multimedia*, 2004, pp. 472–480.
[19] C.-C. Chang and C.- J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. on Intelligent Syst. and Technol.*, pp. 2:27:1–27:27, 2011.
[20] S. M. Kay, *Modern Spectral Estimation: Theory and Application*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1998.
[21] G. Cao, Y. Zhao, R. Ni, and A. Kot, "Unsharp masking sharpening detection via overshoot artifacts analysis," *IEEE Signal Process. Lett.*, vol. 18, no. 10, pp. 603–606, Oct. 2011.
[22] S. Bayram, I. Avcubas, B. Sankur, and N. Memon, "Image manipulation detection," *J. Electron. Imag.*, vol. 15, no. 4, pp. 04110201–4110217, 2006.
[23] X. Kang, M. C. Stamm, A. Peng, and K. J. R. Liu, "Robust median filtering forensics based on the autoregressive model of median filter residual," in *Proc. APSIPA Annu. Submit Conf. 2012*, Los Angles, CA, USA, Dec. 2012.
[24] C. Chen and J. Ni, "Median filtering detection using edge based prediction matrix," in *Proc. IWDW 2011*, Atlantic City, NJ, USA, Dec. 2011.
[25] BOSS [Online]. Available: http://exile.felk.cvut.cz/boss/BOSS-Final/index.php?mode=VIEW&tmpl=materials
[26] P. Bas and T. Furon, Break Our Watermarking System [Online]. Available: http://bows2.ec-lille.fr/ 2nd
[27] T. Gloe and R. Böhme, "Dresden Image Database for benchmarking digital image forensics," in *Proc. 2010 ACM Symp. on Appl. Computing*, Sierre, Switzerland, Mar. 22–26, 2010, pp. 1584–1590.
[28] Natural Resources Conservation Service Photo Gallery, United States Department of Agriculture [Online]. Available: http://photogallery.nrcs.usda.gov

Xiangui Kang (M'00) received the B.S., M.S., and Ph.D. degrees from Peking University in 1990, Nanjing University in 1993, and Sun Yat-Sen University, Guangzhou, China, in 2004, respectively.

Dr. Kang is currently a professor with the School of Information Science and Technology, Sun Yat-sen University, Guangzhou, China. He visited the University of Maryland, College Park, MD, USA from August 2011 to August 2012, and New Jersey Institute of Technology from August 2004 to September 2005. His research interests include information forensics and security, game theory, multimedia communications and security. He and his supervised student received the Best Student Paper Award from the International Workshop on Digital-Forensics and Watermarking in 2008. He is a member of the IEEE ComSoc's Multimedia Communications Technical Committee and APSIPA image, video, and multimedia technical committee. He serves as the associate editor of the *KSII Transactions on Internet and Information System*.

Matthew C. Stamm (S'08–M'12) received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of Maryland, College Park, MD, USA, in 2004, 2011, and 2012, respectively. He is currently a Postdoctoral Research Associate with the Department of Electrical and Computer Engineering at the University of Maryland, College Park. He will join Drexel University as an Assistant Professor in the Department of Electrical and Computer Engineering in August 2013. His research interests include signal processing and information security with a focus on digital multimedia forensics and anti-forensics.

Dr. Stamm received the Dean's Doctoral Research Award in 2012 from the A. James Clark School of Engineering at the University of Maryland. Additionally, he received a Distinguished Teaching Assistant Award in 2006, a Future Faculty Fellowship in 2010, and the Ann G. Wylie Fellowship in 2011 from the University of Maryland. From 2004 to 2006, he was a Radar Systems Engineer with the Johns Hopkins University Applied Physics Laboratory.



Anjie Peng received the M.A. degree from the School of Mathematics and Computational Science, Sun Yat-sen University, Guangzhou, China. He is currently working toward the Ph.D. degree at the School of Information Science and Technology, Sun Yat-sen University, Guangzhou, China. His research interests include multimedia forensics and machine learning.

K. J. Ray Liu (F'03) was named a Distinguished Scholar-Teacher of the University of Maryland, College Park, MD, USA, in 2007, where he is Christine Kim Eminent Professor of Information Technology. He leads the Maryland Signals and Information Group conducting research encompassing broad areas of signal processing and communications with recent focus on cooperative and cognitive communications, social learning and network science, information forensics and security, and green information and communications technology.

Dr. Liu is the recipient of numerous honors and awards including IEEE Signal Processing Society Technical Achievement Award and Distinguished Lecturer. He also received various teaching and research recognitions from the University of Maryland including university-level Invention of the Year Award, Poole and Kent Senior Faculty Teaching Award, Outstanding Faculty Research Award, and Outstanding Faculty Service Award, all from A. James Clark School of Engineering. An ISI Highly Cited Author, Dr. Liu is a Fellow AAAS.

Dr. Liu is President of IEEE Signal Processing Society where he has served as Vice President–Publications and Board of Governor. He was the Editor-in-Chief of *IEEE Signal Processing Magazine* and the founding Editor-in-Chief of *EURASIP Journal on Advances in Signal Processing*.