A Multiscale and Multi-Perturbation Blind Forensic Technique for Median Detecting*

Anselmo Ferreira and Anderson Rocha

Institute of Computing, University of Campinas
Av. Albert Einstein, 1251, Cidade Universitaria, Campinas/SP - Brasil,
CEP 13083-852
{anselmoferreira,anderson.rocha}@ic.unicamp.br

Abstract. This paper aims at detecting traces of median filtering in digital images, a problem of paramount importance in forensics given that filtering can be used to conceal traces of image tampering such as resampling and light direction in photomontages. To accomplish this objective, we present a novel approach based on multiple and multiscale progressive perturbations on images able to capture different median filtering traces through using image quality metrics. Such measures are then used to build a discriminative feature space suitable for proper classification regarding whether or not a given image contains signs of filtering. Experiments using a real-world scenario with compressed and uncompressed images show the effectiveness of the proposed method.

Keywords: Median Filtering, Image Tampering, Image Forensics.

1 Introduction

The ever-growing availability of digital images resulting from the massive use of cheap digital cameras and social networks has allowed people to easily share information all over. With such a massive flood of information the question is how to tell the real from the fake. The old adage: "one image is worth a thousand words" no longer holds intact. Frequently people are mislead and innocently believe in what photographs depict. Therefore the development of reliable forensic detection tools are paramount.

One form of detecting the presence of image tampering is through the analysis of artifacts left by the resampling operations. Popescu and Farid [1] noted that re-sampling operations use interpolation techniques (which results in one image with pixels correlated in some way) and proposed an Expectation-Maximization technique for finding periodic samples of the image and detecting resampling operations. One particular problem of this technique is the assumption of a linear correlation of the pixels. As stated by Kirchner and Bohme [2], a non-linear filter such as the median filter can destroy these re-sampling artifacts by replacing each

* The authors thank the financial support of CNPq (Grants #477662/2013-7, and #304352/2012-8), FAPESP (Grant #2010/05647-4), and Microsoft Research.

E. Bayro-Corrochano and E. Hancock (Eds.): CIARP 2014, LNCS 8827, pp. 302–310, 2014.

pixel with the median-valued pixel within a neighborhood. Median filtering can also be used to fool some light direction-based forensic techniques such as the one proposed by Johnson and Farid [3] and its extension by Saboia et al [4]. Given the use of median filtering operations for doctoring and hiding traces of image doctoring, several researchers have tackled the problem of detecting it over the last years [5–12]. Most of these methods assume that the median filtering has *streaking* artifacts and use them as proxy to detect filtering.

In this paper, we propose a median filtering detection algorithm based on the hypothesis that the median filtering *streaking* artifacts affect the image quality under multiscale filterings (filterings with different regions of interest) and over progressive perturbations (henceforth perturbations are defined as cascade-wise successive image filterings). We evaluate image quality metrics upon perturbed images building a highly discriminative feature space for future classification. Experiments with compressed and uncompressed public datasets confirm the method's competitiveness without assuming anything about the underlying filtering process of the input images.

2 State of the Art

Most of the median filtering detection techniques proposed in the literature rely on artifacts known as *streaking* [13]. In median filtered images, the probability of 2 pixels in a given distance have the same value is high due to the nature of the median filtering: when the sliding window moves, the probability of the changed pixel being the median value of the new pixel grid in the translated window is high [5].

Kirchner and Fridrich [5] proposed a method based on a histogram of differences between an image and its version translated one pixel. In median filtered images, the *streaking artifacts* refer to pixels with the same value. The ratio of bin related to 0 value (h_D^0) and the adjacent bins (h_D^1, h_D^{-1}) are higher in median filtered images than this same ratio in pristine (non-filtered) images. To detect the median filtering, the authors proposed the use of an ad-hoc threshold. Cao et al [6] also explored the streaking artifacts by computing the first order of pixel differences with the upper neighbor of a pixel in an image I and binarize these differences. The authors then build two matrices of differences, one containing the first order of neighboring pixel differences calculated considering only the pixels in rows and another containing pixels column-wise. As the pixels have similar values in a given neighborhood due to the streaking artifacts, the variance in a squared region around each pixel is low.

Yuan [7] stated that the streaking artifacts yield dependencies in overlapped neighboring blocks of pixels. To detect this, the author proposed a set of metrics to be calculated in each pixel in each $s \times s$ non-overlapped blocks. Chen and Ni [8] created an Edge-Based Prediction Matrix (EBPM) of different kinds of edges and used it to yield 72 dimensional feature vectors used to differentiate median and pristine images in a classifier.

Chen et al. [9, 10] stated that median filtered images inevitably exhibit distinctive statistical artifacts in the difference domain. They studied the Cumulative

Distribution Function of the first order differences of pixels and proposed an approach which yields 56 features used by an SVM classifier, 44 of them based on a global probability feature set and 12 based on a Local Correlation Feature Set.

Kang et al. [11, 12] applied an autoregressive model to the Median Filtered Residual (MFR), which is the difference between the filtered image and the original image. This autoregressive model yields 10 coefficients used to feed an SVM classifier.

Differently to all of the aforementioned approaches, the proposed method works by exploring the effects of multiple and multiscale perturbations, using image filtering to highlight streaking artifacts and building a highly discriminative feature space suitable for automatic decision making with image quality metrics. We give more details of the proposed method in the next section.

3 Proposed Method

For detecting median filtered images, we observe that an image that was never median-filtered, when filtered for the first time, will exhibit a behavior different from the already filtered images after the same operation. We can make an analogy to text file compression in this case. When a text file is compressed for the first time, most often the file size decreases because there are redundancies in the text. However, when a second compression is applied to an already compressed file, chances are the file size will increase. This happens because there are much less redundant elements to compress, so the file size will increase compared to the previous compression.

An opposite behavior happens in median-filtered images after filtering them again. In this case, the redundancy will increase because the *streaking artifacts* will be highlighted. This can be easily observed if one applies median filtering until idempotency. To detect such behavior, we perform n progressive and multi-scale perturbations on the image. These perturbations consist of multi-scale filters applied on the image progressively. We consider filtering windows of size 3×3, 5×5, 7×7 and 9×9.

The rationale for using the multi-scale perturbations is that when an already filtered image is filtered once more using a different median filtering window, the *streaking artifacts* will be emphasized. When applied in succession (progressively) it tends to find groups of streaking pixels instead of just a few of them when considering just one scale as in previous approaches. By applying the median filtering in this way, the image quality will degrade differently from pristine images, so image quality metrics can capture the traces of median filtering after the progressive and multiscale perturbations.

Image Quality Metrics [14–16] have been successfully employed in the literature to calculate how a given image was distorted according to a reference image. They are useful when imaging systems introduce distortions or artifacts in the signal caused by motion blurring and sensor inadequacy. Given an input image, we compare it to its multiple perturbation filtered version by using eight

bivariate image quality metrics per perturbation and scale: Peak Signal to Noise Ratio, Structural Content, Average Difference, Maximum Difference, Normalized Cross Correlation, Normalized Absolute Error, Structural Similarity, and Mean Squared Error.

The proposed method performs a set of perturbations by applying median filtering n times with $w \times w$ window sizes and calculating q quality metrics on each filtered image version. The filtered image is the input for the following filtering and we can use s windows (scales) with different sizes to perform the filtering. By comparing each filtering result with the input image using q quality metrics, we can build a discriminative feature space for proper learning of a classifier by concatenating the q quality images calculated per filtered image. The feature vector has therefore $s \times n \times q$ dimensions. The same is done to testing images and a classifier (such as Support Vector Machines) trained with the data from the training images can discriminate between the pristine and median filtered images. Figure 1 shows how the proposed method works.

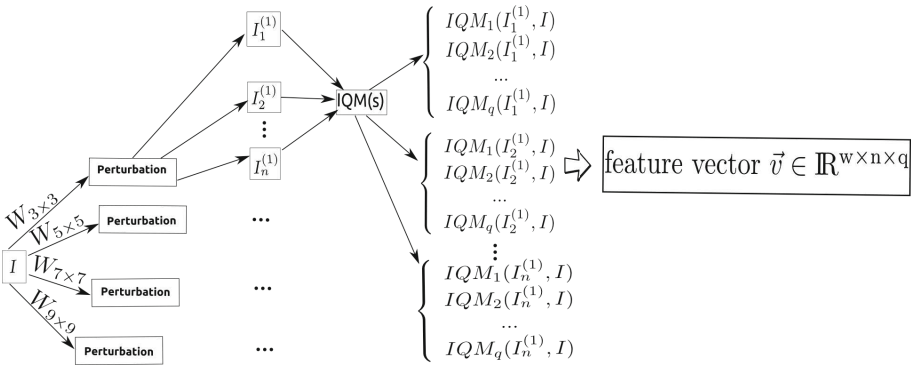


Fig. 1. An image is filtered n times in different scales ($s = 4$) and eight quality metrics are calculated ($q = 8$). The feature space comprises the fusion of $n \times 8$-d vectors created at each perturbation in four scales for each image. Any machine learning classifier can later be used to carve the decision boundary in the formed IQMs feature space.

Our method resembles the one by Rocha and Goldenstein [17] in which they propose a steganalysis algorithm based on progressive embeddings (progressive insertion of hidden messages) on digital images. Avcibas et al. [18] used before IQMs to detect traces of image tampering. Previously, the authors had already used such technique to perform image steganalysis [19]. Our contribution is the use multiple and progressive perturbations allied with image quality metrics on images to detect median filtering as we observed this process is extremely effective for detecting streaking artifacts.

4 Experiments

For validating the proposed method and comparing it to the ones in literature, we have chosen a *Cross-Dataset* validation protocol in which we have training samples containing images in similar conditions and testing samples with a variety of conditions which can be seen as a real-world situation.

The training datasets comprise images with almost similar lighting conditions, low resolution and the median filtered samples were blurred using one median filtering window size (i.e., 3×3). The compressed images training set contains 3,996 JPEG images from the *Chinese Academy Image Tampering Database* (CASIA) [20].

The compressed images testing dataset is more complex and comprises 800 JPEG images collected with very different resolutions, taken from different cameras and smartphones. Also, the blurred images in the testing dataset were blurred with different median filtering implementations (MATLAB, OPENCV and GIMP) and different median window sizes. We used a similar configuration for a situation where compressed and uncompressed images can occur. In this case, we used the previous compressed dataset along with 2,773 uncompressed images for training from CASIA [20] and 1,338 uncompressed images from UCID [21] for testing. All datasets are freely available at their original websites and will also be at `http://tinyurl.com/nsxe8j8` upon acceptance along with the necessary materials for reproducing all the experiments.

The experiments have two parts. First, we assess the choices for improving the proposed classifier such as the number of considered scales and perturbations for building up the feature vectors. Then the best configuration found during this validation step is used to compare it to existing methods in the literature (second part) considering compressed and uncompressed images.

4.1 Part #1: Choosing the Method's Parameters

The minimal number n of perturbations of the proposed technique was found in statistical tests after randomly splitting the 3,996 images of the publicly-available CASIA [20] dataset 10 times in training and testing sets (also known as 5×2 cross-validation).

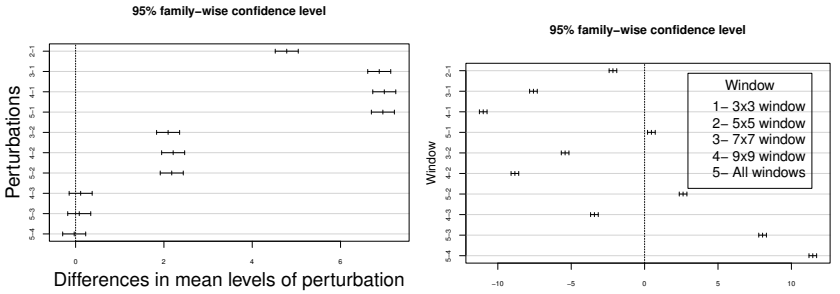
Since we are using this dataset for fine tuning the parameters of our method, it will be used only as a training dataset in Sec. 4.2 when comparing our method to others in the literature. That is why we chose to perform a cross-dataset validation being fair and closer to the actual forensic scenario one might face. Here, we characterize the images as described in Sec. 3 and train an SVM classifier with an RBF kernel from LibsVM [22], whose parameters are automatically learned during training according to the 5×2 cross-validation protocol used.

For validation, we performed two rounds of experiments: in the first one we used just one window $w \times w$ for blurring, where $w \in \{3, 5, 7, 9\}$ and vary the number of perturbations n in the interval $1 \leq n \leq 5$. In the second round, we used a multiscale approach, where we use all of the four window sizes for filtering the images and vary only the perturbations. Table 1 shows the three best experiment results.

Table 1. Results in mean classification accuracy after a 5 × 2 cross-validation on CASIA dataset [20]

Number of Perturbations	Windows	Result
3	3 × 3	98.8% ± 0.22
4	3 × 3, 5×5, 7×7, 9×9	98.7% ± 0.29
3	3 × 3, 5×5, 7×7, 9×9	98.7% ± 0.28

The ANOVA statistical test results (not shown) confirm that varying the number of windows and perturbations are statistically significant (p-value $<$ 0.05) and these factors are correlated. Figure 2 show the results of Tukey tests for pairwise comparisons.



(a) (b)

Fig. 2. (a) Tukey test pairwise comparison in factor perturbation (b) Tukey test pairwise comparison in factor window

As we can see on Fig. 2(a), there is no statistical difference when using three and four, three and five and four and five perturbations. However, there is significative difference when using more than one perturbation. In addition, varying the window sizes is statistically significant according to Fig. 2(b). The ANOVA test in the three best algorithms yielded a p-value of 0.79, which helps us to state that the accuracy difference between these techniques is not statistically significant. We chose to use the two last configurations in the second part of the experiments when comparing to the methods in the literature because the first one yielded slightly worse results.

4.2 Part #2: Comparison with State-of-the-Art Methods

We now turn our attention to comparing the classification results of two of the three best approaches of the proposed technique with 128 and 96 dimensional feature vectors. We call these techniques, respectively, as FPMW (Four Perturbations, Multiple Windows) and TPMW (Three Perturbations, Multiple Windows). We compare them to state-of-the-art methods (Sec. 2): (1) Kirchner

Table 2. Compressed dataset experiments results and McNemar's statistical tests comparing the best technique (FPMW) with others. TPMW and FPMW are variations of the proposed method discussed in Sec. 3.

	TPMW	FPMW	SPAM [5]	MFF[7]	GLF[9, 10]
Accuracy	82.1%	84.5%	70.1%	70.1%	65.1%
Sensitivity	92%	91%	98 %	88%	99%
Specificity	72%	77%	42%	52%	31%
Precision	76%	80%	62%	64%	59%
Significant?	yes	-	yes	yes	yes

Table 3. Cross dataset experiment with compressed and uncompressed images and McNemar's statistical tests comparing the best technique (TPMW) with others. TPMW and FPMW are variations of the proposed method discussed in Sec. 3.

	TPMW	FPMW	SPAM [5]	MFF[7]	GLF[9, 10]
Accuracy	82.2%	80.8%	77.9 %	74.2%	79.9%
Sensitivity	78.2%	74.4 %	68.3%	76.9%	90.9%
Specificity	90.2%	91.5%	90.6 %	78.6%	76.8%
Precision	88.9%	89.8%	87.9 %	78.3 %	79.7%
Significant?	-	yes	yes	yes	yes

and Fridrich [5] with $T = 3$ and second order Markov Chains as described in their work, yielding a 686-d feature vector (we call it SPAM), (2) Yuan [7] in 3×3 blocks and a 44-d feature vector (which we call MFF) and (3) Chen et al. [9, 10] with parameters T=10, B=3 and K=2 and 56-dimensional feature vectors as described in their work (which we call GLF).

We then used an SVM with RBF kernel from LibsVM [22] whose parameters were learned during training using the LibSVM's built-in grid-search method. Table 2 shows the results for the compressed dataset and the McNemar's statistical test results between the best ranked technique and the others. Table 3 shows tests for the scenario with compressed and uncompressed images present during training and testing.

According to Table 2, the proposed technique FPMW presents the best accuracy by correctly classifying 676 out of 800 test images while SPAM and GLF correctly classified 561 out of 800 images and 521 out of 800 images, respectively. Note that both SPAM and GLF present low specificities (42% and 31% respectively) in this dataset. In a forensic scenario, low specificity often means putting the blame on an innocent person and is undesirable. The second best technique was the proposed TPMW, which correctly classified 657 out of 800 images followed by MFF (561 out of 800 images).

Table 3 shows the proposed method performance compared to the ones in literature when dealing with compressed and uncompressed images simultaneously. In this case, TPMW presents the best accuracy correctly classifying 1,801 out of 2,138 images (remember that the test dataset now contains 800 compressed and 1338 uncompressed images). In sequence, the next top performers are FPMW,

GLF, and SPAM. MFF presented the worst results (1,663 images correctly classified). Although the proposed methods perform well in these scenarios, there are situations where they are not top performers. This happens specially when the training and testing sets contain only images without any compression. Considering a cross-dataset scenario where only uncompressed images are used for training and testing, the proposed methods present smaller accuracies (TPMW with 87.5%, FPMW with 87.8%) than its counterparts (GLF with 99.7%, MFF with 99.9% and SPAM with 99.7%). This is not a problem since in practice it is unlikely that only training and testing images would be available.

5 Conclusion

In this paper, we presented a novel approach to forensically detect median blurring traces on digital images. Our technique is different from others as it progressively perturbs the image by blurring it multiple times with different window sizes (filtering intensities), building a discriminative feature for later decision making by using image quality metrics.

Upon training an SVM classifier on such feature space, the method can automatically find traces of filtering even with a simplified training set. We showed the method's reliability when tested with diverse images on a cross-dataset protocol which we believe to be a more real-world situation. The experiments and results corroborate our initial hypothesis that multiple perturbations with different intensities capture the telltales left behind by median filtering algorithms in a competitive way with some existing methods.

As future work, we will focus our study on more image quality metrics that can be combined to the proposed feature vector, include more median blurring intensities in the dataset, study the proposed approach on different image compression settings, perform fusion of classifiers and study the proposed method under the median filtering anti-forensic operation [23].

References

1. Popescu, A.C., Farid, H.: Statistical tools for digital forensics. In: Fridrich, J. (ed.) IH 2004. LNCS, vol. 3200, pp. 128–147. Springer, Heidelberg (2004)
2. Kirchner, M., Bohme, R.: Hiding traces of resampling in digital images. IEEE Trans. on Inf. For. and Sec. 3, 582–592 (2008)
3. Johnson, M.K., Farid, H.: Exposing digital forgeries through specular highlights on the eye. In: Furon, T., Cayre, F., Doërr, G., Bas, P. (eds.) IH 2007. LNCS, vol. 4567, pp. 311–325. Springer, Heidelberg (2008)
4. Saboia, P., Carvalho, T., Rocha, A.: Eye specular highlights telltales for digital forensics: a machine learning approach. In: Intl. Conference on Image Processing, pp. 1977–1980 (2011)
5. Kirchner, M., Fridrich, J.: On detection of median filtering in digital images. In SPIE Media Forensics and Security II, pp. 754110-754110-12 (2010).
6. Cao, G., Zhao, Y., Ni, R., Yu, L., Tian, H.: Forensic detection of median filtering in digital images. In: IEEE Intl. Conference on Multimedia & Expo, pp. 89–94 (2010)

7. Yuan, H.D.: Blind forensics of median filtering in digital images. IEEE Trans. on Infor. For. and Sec. 6, 1335–1345 (2011)
8. Chen, C., Ni, J.: Median filtering detection using edge based prediction matrix. In: Shi, Y.Q., Kim, H.-J., Perez-Gonzalez, F. (eds.) IWDW 2011. LNCS, vol. 7128, pp. 361–375. Springer, Heidelberg (2012)
9. Chen, C., Ni, J., Huang, R., Huang, J.: Blind median filtering detection using statistics in difference domain. In: Kirchner, M., Ghosal, D. (eds.) IH 2012. LNCS, vol. 7692, pp. 1–15. Springer, Heidelberg (2013)
10. Chen, C., Ni, J., Huang, J.: Blind detection of median filtering in digital images: A difference domain based approach. IEEE Trans. on Im. Proc. 22, 4699–4710 (2013)
11. Kang, X., Stamm, M., Peng, A., Liu, K.: Robust median filtering forensics using an autoregressive model. IEEE Trans. on Infor. For. and Sec. 8, 1456–1468 (2013)
12. Kang, X., Stamm, M., Peng, A., Liu, K.: Robust median filtering forensics based on the autoregressive model of median filtered residual. In: IEEE Signal Information Processing Association Annual Summit and Conference, pp. 1–9 (2012)
13. Bovik, A.: Streaking in median filtered images. IEEE Trans. on Acous. Sp. and Sig. Proc. 35, 493–503 (1987)
14. Thung, K., Raveendran, P.: A survey of image quality measures. In: IEEE Intl. Conference for Technical Postgraduates, pp. 1–4 (2009)
15. Eskicioglu, A., Fisher, P.: Image quality measures and their performance. IEEE Trans. on Comm. 43, 2959–2965 (1995)
16. Wang, Z., Bovik, A., Sheikh, H.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. on Im. Proc. 13, 600–612 (2004)
17. Rocha, A., Goldenstein, S.: Progressive randomization: Seeing the unseen. Elsevier Comput. Vis. and Im. Underst. 114, 349–362 (2010)
18. Avcibas, I., Bayram, S., Memon, N., Ramkumar, M., Sankur, B.: A classifier design for detecting image manipulations. In: IEEE Intl. Conference on Image Processing, pp. 2645–2648 (2004)
19. Avcibas, I., Memon, N., Sankur, B.: Steganalysis based on image quality metrics. In: IEEE Workshop on Multimedia and Signal Processing, pp. 517–522 (2001)
20. Casia tampered image detection database, `http://forensics.idealtest.org/`
21. Schaefer, G., Stich, M.: Ucid - an uncompressed colour image database. In: Storage and Retrieval Methods and Applications for Multimedia, pp. 472–480 (2004)
22. Chang, C., Lin, C.: LIBSVM: A library for support vector machines. ACM Trans. on Intell. Syst. and Tech. 2, 27:1-27:27 (2011), `http://www.csie.ntu.edu.tw/~cjlin/libsvm`
23. Fontani, M., Barni, M.: Hiding traces of median filtering in digital images. In: European Signal Processing Conference, pp. 1239–1243 (2012)