

SPATIAL DOMAIN QUANTIZATION NOISE BASED IMAGE FILTERING DETECTION

Hareesh Ravi[†], A.V. Subramanyam[†], Sabu Emmanuel[‡]

[†] Electronics and Communication Engineering, Indraprastha Institute of Information Technology, India

[‡] Computer Science and Engineering, Kuwait University, Kuwait

ABSTRACT

Smart image editing and processing techniques make it easier to manipulate an image convincingly and also hide any artifacts of tampering. Common real world forgeries can be accompanied by enhancement operations like filtering, compression and/or format conversion to suppress forgery artifacts. Out of these enhancement operations, filtering is very common and has received a lot of attention in forensics research lately. However, different filtering operations and image formats are not investigated deeply and simultaneously. We propose an algorithm to detect if a given image has undergone filtering based enhancement irrespective of the format of image or the type of filter applied. In the proposed algorithm, we exploit the correlation of spatial domain quantization noise of an image by extracting transition probability features and classify the image as filtered or unfiltered. Experiments are performed to evaluate the robustness and compare the performance of the proposed technique with popular forensic filtering detection algorithms and is found to be superior in most of the cases.

Index Terms— Filtering detection, Quantization noise, Markov features, JPEG and TIFF.

1. INTRODUCTION

Assuring the integrity/authenticity of digital evidence images is of paramount importance for Law Enforcement Agencies. Several techniques have been developed to detect if a given image is authentic or forged. Since a forgery is often followed by an enhancement technique to make the forgery less detectable and more convincing, recently algorithms are being developed to detect such enhancement operations. Operations like noise reduction, filtering, contrast enhancement, de-blurring and edge sharpening etc can be performed as part of the forgery [1]. Therefore, detecting these enhancement operations that manipulate the image, can be considered a valid assumption that indicates forgery. In [2] - [6], double compression is considered as a possible indication of forgery. This may not be true always, as an image may be simply decompressed and recompressed again or saved in a different format without any other manipulation.

In view of the effect of filtering on forgery or forensic algorithms, authors of [7], [8] and [9] proposed the importance of median filtering detection and how non-linear filtering operations can be used maliciously to hide fingerprints left by forgery. In order to counter this problem, techniques like [7] - [11] were proposed to detect median filtering specifically. These techniques, however, do not consider other linear filtering operations that can indicate possible forgery. For example, in [8] authors have specifically used the coefficients of an Auto Regressive model by fitting the Median Filter Residual (MFR) of each image. The same features or model cannot be used to detect linear filtering. Moreover, common scenarios such as filtering of JPEG compressed images or double compressed images are not investigated. Similarly, linear filtering detection algorithms like [1] and [12] proposed to detect and classify full frame lin-

ear filtering and compression. In both these papers, however, JPEG compression post filtering is not considered. This might suppress the artifacts of operations that the image underwent prior to compression [8]. Also, the effect of median filtering in JPEG images and filtering in uncompressed images are not discussed. The features used in the paper are specific to the effect of linear filtering on DCT histograms and hence the same features or model may not be effective in detecting median filtered images. In [13] authors considered various kinds of manipulations such as filtering, contrast enhancement etc. obtaining a good accuracy using Fusion boost ensemble classifier. However, results for lower quality factors of JPEG compression are not provided. Also the data set considered is small. It is unrealistic

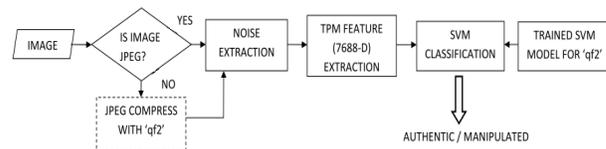


Fig. 1. Block diagram of the forensic analysis pipeline for authentication

for practical applications to have different algorithms for each type of filter, format of image, type of forgery etc as we may never know which one to use. Therefore, it is necessary to come up with algorithms that can consider a wide possibility of scenarios and formats of images. For example, in case of filtering detection, the algorithm should be able to detect if a given image is filtered or unfiltered, irrespective of whether the image is compressed or uncompressed. In addition, both linear and non linear filtering should be considered. Hence, we propose an efficient technique to detect images that are enhanced using linear filters (Gaussian, Laplacian, Average, Unsharp) and non linear filters (Median). In our experiments both JPEG and TIFF¹ images are considered.

We believe this is the first forensic algorithm that targets both linear and non linear filters for different image formats. The block diagram of a typical forensic pipeline in this case is given in Fig 1. To replicate real world manipulation process, we also perform common forgeries such as splicing and copy move along with filtering to show they do not affect the efficiency of the filtering detection algorithm. The technique is based on the principle that, when filtering and compression are applied, the spatial correlation of the compression noise in the image gets perturbed. Spatial domain compression noise is shown to be correlated by Robertson et. al. in [14]. It is shown in Fig 2, that compression noise of unfiltered images has low pass characteristics and when filtering (low pass, high pass or median) is applied, this correlation is disturbed. We leverage this behavior of compression noise to propose a single model that can detect if a given image is filtered or unfiltered for the considered filters. We use the quantization noise model proposed by [14] and a modification of the natural image model proposed in [15] to extract compression

¹JPEG compression is lossy while TIFF is uncompressed.

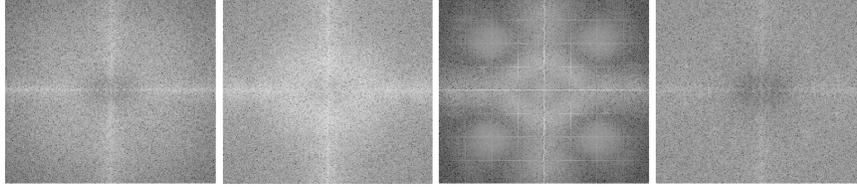


Fig. 2. Average Power spectral density obtained from compression noise of 650 unfiltered, low pass, median and high pass (left to right) filtered images.

noise. Though other image prior models can be used, we propose a modification to this prior model that provides favorable results (Sec 4.1) by incorporating the effects of blocking artifacts produced during compression. The noise thus extracted is then modelled as a first order spatial ergodic Markov chain which has been proven to be an effective feature ([16] - [19]). These features are used to detect whether a given image has been filtered or unfiltered. The results are provided using standard UCID [20], NCID [21] and Dresden[22] image databases.

The rest of the paper is organized as follows. Section 2 gives the related works and the proposed modified prior model based noise extraction followed by feature extraction. In Section 3, the experimental setup and database creation is given. Section 4 provides the detection results and evaluates the efficiency of our method with the existing algorithms. Section 5 concludes the paper by providing future works.

2. PROPOSED SCHEME

Let \mathbf{I} be a natural JPEG image scene in spatial domain under consideration. Let \mathbf{Z} be the vectorized form of an 8×8 block of the image \mathbf{I} in spatial domain while \mathbf{Y} the vectorized form of the corresponding block in DCT domain. All details and variables here after are given for a single block, unless specified otherwise. The compression noise for a non overlapping 8×8 block can be written as $\mathbf{e}_z = \mathbf{Z} - \mathbf{Z}_q$ where \mathbf{Z}_q is the quantized block. This noise is characterized as a zero mean multivariate Gaussian with \mathbf{K}_{e_z} as its covariance matrix as,

$$\mathbf{P}(\mathbf{e}_z) = \frac{1}{(2\pi)^{D/2} |\mathbf{K}_{e_z}|^{1/2}} \exp\left(-\frac{1}{2} \mathbf{e}_z^T \mathbf{K}_{e_z}^{-1} \mathbf{e}_z\right) \quad (1)$$

where $D = 64$ is the number of dimensions of the multivariate Gaussian². The natural image prior model used in our algorithm is obtained from [15]. This model is based on the Markov Random Field (MRF) modeling [15] of images that represents the local structures of an image using conditional probability distribution where image pixels' values depend only on its neighborhood. The joint distribution is given by Gibbs measure as,

$$p(\mathbf{Z}) = \frac{1}{\beta} \exp\left(-\lambda \sum_{c \in C} \rho_T(\mathbf{d}_c^t \mathbf{Z})\right) \quad (2)$$

where, β is a normalization constant, c are local groups called cliques whereas C is the set of all such cliques depending on the neighborhood structure of the Huber Markov Random Field (HMRF) [14]. Here, $\rho_T(\cdot)$ is the Huber function [15] and vectors \mathbf{d}_c^t extracts differences between a pixel and its neighbors such that the prior model degenerates to,

$$p(\mathbf{Z}) = \frac{1}{\beta} \exp\left(-\lambda \sum_{m=0}^{M-1} \sum_{n \in N_m} \rho_T(\mathbf{Z}[n] - \mathbf{Z}[m])\right) \quad (3)$$

Where N_m is the index set of neighbors for the m^{th} pixel, and M is the number of pixels in the block. We next present the denoising algorithm using a modified version of the above mentioned prior model followed by feature extraction and classification.

2.1. Quantization Noise Extraction

We modify the prior model given in eq (2) by introducing a multiplicative factor w in the Huber function to incorporate the effect of blocking artifacts caused due to compression. Thus, the new Huber function $\rho_T(\cdot)$ after modification is,

$$\rho_T(u) = \begin{cases} wu^2, & |u| \leq T, \\ w(T^2 + 2T(|u| - T)), & |u| > T \end{cases} \quad (4)$$

$$w = \begin{cases} 1 & \forall Z(u) : u \notin \mathcal{S}, \\ \gamma & \text{otherwise} \end{cases}$$

where T the threshold set in [14]. The parameters γ (hence w for boundary pixels) and T are empirically determined. \mathcal{S} is the set of pixels which belong to the border pixels in each 8×8 block. We use Bayesian MAP estimation for extracting the compression noise from the prior and the noise model. The maximum *a posteriori* (MAP) criterion is,

$$\hat{\mathbf{Z}} = \arg \max_{\mathbf{Z}} p(\mathbf{Z} | \mathbf{Z}_q) = \arg \max_{\mathbf{Z}} p(\mathbf{Z}) p(\mathbf{Z}_q | \mathbf{Z}) \quad (5)$$

where $\hat{\mathbf{Z}}$ is the final estimate for the block after removing the compression noise. Though \mathbf{Z}_q is a deterministic function of \mathbf{Z} , the $p(\mathbf{Z}_q | \mathbf{Z})$ term in the above equation is considered as a Gaussian random variable with mean \mathbf{Z} and auto covariance \mathbf{K}_{e_z} as explained in [14]. After substituting eq (1) and eq (3) in eq (5) we get,

$$\hat{\mathbf{Z}} = \arg \max_{\mathbf{Z}} \left\{ \frac{1}{\beta} \exp\left(-\lambda \sum_{m=0}^{M-1} \sum_{n \in N_m} \rho_T(\mathbf{Z}[n] - \mathbf{Z}[m])\right) \left(\frac{1}{(2\pi)^{D/2} |\mathbf{K}_{e_z}|^{1/2}} \exp\left(-\frac{1}{2} \mathbf{e}_z^T \mathbf{K}_{e_z}^{-1} \mathbf{e}_z\right) \right) \right\} \quad (6)$$

In order to maximize eq (5), the argument of $\exp(\cdot)$ in eq (6) is minimized using gradient descent algorithm³. Combining all the resulting denoised non overlapping blocks $\hat{\mathbf{Z}}$, we generate the denoised image $\hat{\mathbf{I}}$. The compression noise \mathbf{N}_c for the image is then obtained as $\mathbf{N}_c = \mathbf{I} - \hat{\mathbf{I}}$.

²Please refer [14] for a detailed explanation.

³Details of the algorithm are not given owing to space constraints

CLASS	IMAGE CAPTURED (No. Of Images)	OPERATIONS BY ADVERSARY			FORENSIC END
		Manipulation	Enhancement	Saved as	
UNFILTERED	TIFF (1000) & JPEG- qf_1 (2000)	–	–	TIFF/JPEG- qf_2	JPEG (qf_2)
FILTERED	TIFF (1500) & JPEG- qf_1 (1500)	CM/SP	filtering from Table 2	TIFF/JPEG- qf_2	JPEG (qf_2)

Table 1. Data set creation

2.2. Transition Probability Feature Extraction

The noise \mathbf{N}_c extracted from a JPEG compressed image can be modeled as a first order ergodic spatial Markov chain such that, $p(X_{t+1} = x | X_1 = x_1, X_2 = x_2, \dots, X_t = x_t) = p(X_{t+1} = x | X_t = x_t)$, where X_{t+1} is the present state and (X_1, X_2, \dots, X_t) are the previous states. The features that we use to characterize this noise is Transition Probability Matrix (TPM). The states of the chain are the elements of the difference array (gradient along eight directions) obtained from the absolute value of noise matrix as shown in eq (7). It was found experimentally that eight directions rather than just four as in literature [3, 19], gave a better result. Representation along the right direction is shown hereafter and those along other directions can be obtained in a similar way.

$$\mathbf{D}_c^{\rightarrow}(i, j) = |\mathbf{N}_c(i, j)| - |\mathbf{N}_c(i, j + 1)| \quad (7)$$

where $i = 1, 2, \dots, \mathbf{M}$ and $j = 1, 2, \dots, (\mathbf{N} - 1)$ are indices representing each element in the matrix. Values in each difference array $\mathbf{D}_c^{\rightarrow}$ are rounded off to the nearest integer to get integer values states and then truncated between $-T_r$ to T_r before extracting the transition probabilities,

$$\tilde{\mathbf{D}}_c^{\rightarrow}(i, j) = \begin{cases} -T_r, \mathbf{D}_c^{\rightarrow}(i, j) < -T_r \\ +T_r, \mathbf{D}_c^{\rightarrow}(i, j) > +T_r \\ \mathbf{D}_c^{\rightarrow}(i, j), \text{ otherwise} \end{cases} \quad (8)$$

This provides us with $(2T_r + 1)$ different states to model the Markov chain. In our model T_r is set to 15 as 95% of the values in the difference array of noise are distributed within $[-15, 15]$. Also, experiments were performed with values between 3 to 30 for T_r and it was found that saturation in performance reached after 15. Moreover, for values lower than 15, percentage of values considered as is, dropped significantly, and so did the accuracy. Now, TPM is constructed as,

$$P_{u,v}^{\rightarrow} = Pr(\tilde{\mathbf{D}}_c^{\rightarrow}(i, j + 1) = u | \tilde{\mathbf{D}}_c^{\rightarrow}(i, j) = v) \quad (9)$$

where, $u, v \in [-T_r, T_r]$, and $u, v \in \mathbb{Z}$. Similarly, the probabilities can be obtained for other directions. The size of each TPM will be 31×31 . This gives us $(2T_r + 1) \times (2T_r + 1) = 961$ transition probabilities for each difference array. The TPMs along the eight directions are concatenated to get the final feature which is $7688 \cdot \mathbf{D}^4$, and this remains the same irrespective of the size of the image. These features are then used to train an SVM model and classify a test image as filtered or unfiltered using the trained model as described in the following sections.

3. DATABASE

We consider 1338 uncompressed images from the UCID database [20] along with 1262 random Never compressed images with significant content from the 5150 images of the NCID database [21].

⁴It is to be noted that, low dimensional linear projection of the feature vector to reduce complexity, did not increase the accuracy. Hence the feature set is entirely considered for classification.

FILTER TYPE (NL / L)	KERNEL SIZE	VALUE ⁵	TOTAL
Median filter (NL)	$(3 \times 3, 5 \times 5, 7 \times 7)$	–	3
Gaussian filter (L)	$(3 \times 3, 5 \times 5)$	$\sigma - 0.5, 1$	4
Average filter (L)	$(3 \times 3, 5 \times 5)$	–	2
Laplacian filter (L)	(3×3)	$\alpha - 0.1, 0.2$	2
Unsharp filter (L)	(3×3)	$\alpha - 0.2, 0.4$	2

Table 2. Enhancement techniques performed as part of forgery. Here ‘NL’ indicates NonLinear filter whereas ‘L’ indicates Linear filter.

All the 1338 ucid images are cropped to 256×256 from the center, matching the size of NCID images. We also take 400 single compressed images captured using different digital cameras from the dresden image database [22]. This together makes 3000 authentic images of size 256×256 called the ‘original set’. Another 1000 random NCID images not present in the previous 1262 are single compressed with a random factor $qf_1 \in (30, 80]$, $qf_1 \in \mathbb{Z}_+$. This set is called the ‘splicing set’. We generate two classes of images for our experiments from these sets as given in Table 1. In the experiments, a JPEG/TIFF image is forged, enhanced using a filter randomly from Table 2 and then saved in TIFF/JPEG format following a typical forgery pipeline. When the given final image is in uncompressed TIFF format, it is JPEG compressed, referred in the Table 1 as ‘Forensic end’. The parameters in the table are to be read as follows, $qf_1 \in (30, 80]$ whereas $qf_2 \in \{30, 40, 50, 60, 70, 80, 90\}$, as in the Table 3. CM indicates copy move forgery of size $s \times s$ where $s \in (50, 130]$, $s \in \mathbb{Z}_+$ and SP indicates splicing forgery performed by copying a $s \times s$ patch of an image from ‘splicing set’ on to the image to be spliced. Compression noise and TPM features are extracted as given in section 2.1 and 2.2 from 6000 (3000 unfiltered and 3000 filtered) images in total for each quality factor qf_2 as in the Table 3 after the above experiments.

4. EXPERIMENTAL RESULTS

In our experiments, λ in eq (6) is set to 0.1 while γ and T in eq (4) are set to 5 and 10 respectively. Results for these parameters are given below and these are empirically determined to give the best results. Out of the 6000 images per quality factor, 1500 images from authentic class and 1500 images from filtered class are used for training while the remaining 3000 images are used for testing. We

Quality Factor	30	40	50	60	70	80	90
Accuracy (%)	80.5	82.4	78.5	82	81	85	82.4

Table 3. Detection accuracy for various quality factors

use RBF kernel binary classification SVM from [23] libsvm library. Grid search is performed for determining the parameters that give the best average cross validation (50% training and 50% testing) ac-

⁵Values are parameters used in matlab for specific filters.

curacy which is provided as $(TPR + TNR)/2$ in Table 3 where TPR is the True Positive Rate and TNR is the True Negative Rate.

4.1. Evaluation

The results of our experiment is given in Table 3. It can be observed from the table that the classification accuracy between an unfiltered and a filtered image is on an average above 80% irrespective of the quality factor of the last JPEG compression. In Fig 3 (a) the average ROC curve of classification using the proposed method is plotted by varying threshold of classification for probability scores generated by SVM. It is to be noted that high TPR of 0.8 is achieved for considerable FPR of 0.16 taking into account multiple scenarios together. Also the time taken for extracting compression noise from one image is around 120 seconds for a 256×256 image while TPM extraction and classification takes approximately 1 second in a 4GB RAM intel i3 core processor based CPU using MATLAB. Compression noise extraction takes more time as each 8×8 block is denoised by gradient descent whose iterations and learning rate are empirically set.

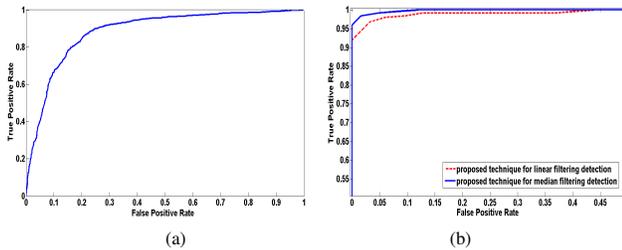


Fig. 3. (a) ROC curve using the proposed method and experimental setup (b) ROC curve using the proposed method for linear filtering and median filtering (refer Sec 4.2).

4.2. Comparison

Since, to the best of our knowledge, there is no work in the literature that simultaneously deals with JPEG and TIFF images along with linear and non linear filtering, we compare the performance of the proposed algorithm in the following manner. We perform separate experiments that adhere to experimental setup and considerations followed in two state of the art techniques. ‘Image manipulation pipeline’ here on means that images in the dataset considered for the corresponding experiments have been through that pipeline of processes.

Firstly, an experiment is done with 1000 images from the ‘original set’ to compare our results with state of the art [1] for linear filtering detection of JPEG images. The pipeline followed is JPEG (qf_2) $\xrightarrow{\text{filtering}}$ TIFF (i.e. dataset consists of filtered JPEG images without any other post processing, since this is the pipeline followed in [1]). To implement our algorithm for this pipeline, forensic compression is done with quality factor 90 since final image of the pipeline is TIFF. Classification is done using 50% samples for training and the rest for testing with SVM using RBF kernel. It is seen from Fig 3 (b) that very high TPR (above 0.85) is achieved under very low FPR (about 0.05) when the proposed method is implemented on experimental setup considered in [1] for $qf_2 = 80$. The accuracy achieved using the proposed algorithm and that of [1] for filtering detection is given in Fig 4. Accuracy is found to be significantly higher for all qf_2 .

Another experiment to detect only ‘median filtering’ is performed using 1000 images to compare with state of the art [7].

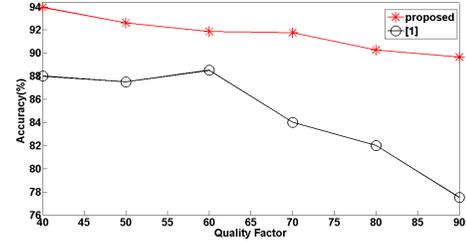


Fig. 4. Comparing detection accuracy with [1] for various compression pairs

Pipeline followed here is TIFF $\xrightarrow{\text{median filter}}$ JPEG (qf_2) (i.e. dataset consists of images that are JPEG compressed post filtering) where $qf_2 = \{70, 90\}$, since only results for these factors are available in [7]. Detection accuracy is given in Table 4 for each qf_2 and median filter kernel as given in [7]. The proposed method gives better or

Median filter size	3 × 3		5 × 5	
Quality factor	90	70	90	70
[7]	98	94.5	98.5	97.5
Proposed	99	95.25	98.5	96.5

Table 4. Detection accuracy for median filtering using [7] and proposed technique

comparable performance in most cases. In Fig 3(b) ROC curve for classification of 5×5 median filtering of JPEG images compressed with quality factor 70 using proposed method is given.

5. CONCLUSION

An effective method to detect filtering based on compression noise characteristics is proposed. The contribution of this method is four fold. *First*, it is not realistic to assume that the given image is JPEG always or it has undergone only one type of filtering. The proposed experimental setup overcomes these constraints by considering compressed and uncompressed images along with multiple types of filters with an average of above 80% accuracy. *Second*, it is to be noted that the method is indifferent to different compression factors and various filters thereby not limiting the application to just double compression, one type of filter or forgery detection. It is observed that two different applications - median filtering detection [7] and linear filtering detection [1] are combined here obtaining good accuracy in unconstrained settings. It also performs better than the state of the art when the settings are constrained as in the literature. *Third*, this technique proves the efficacy of compression noise and paves way for more effective methods using the same to detect manipulation rather than compression. *Finally*, to the best of our knowledge, this is the first approach to solve filtering detection in both JPEG and TIFF images considering various scenarios. In future work, we plan to increase the efficiency of detection and study the effect of localized filters and contrast enhancement on compression noise.

6. ACKNOWLEDGEMENT

The authors thank the ‘Department of Electronics and Information Technology (DeitY)’, Govt. of India, for supporting this work under the project titled ‘Design and Development of Digital Multimedia Forgery Detection System’ with Project No. 12(4)/2014-ESD.

7. REFERENCES

- [1] V. Conotter, P. Comesana, and F. Perez-Gonzalez, "Forensic analysis of full-frame linearly filtered jpeg images," in *Proc. IEEE Int. Conf. on Image Processing*, September 2013, pp. 4517–4521.
- [2] H. Farid, "Exposing digital forgeries from jpeg ghosts," *IEEE Transaction on Information Forensics and Security*, vol. 4, pp. 154–160, March 2009.
- [3] C. Chen, Y.Q. Shi, and W. Su, "A machine learning based scheme for double jpeg compression detection," in *Proc. IEEE Int. Conf. on Pattern Recognition*, December 2008, pp. 1–4.
- [4] F. Huang, J. Huang, and Y.Q. Shi, "Detecting double jpeg compression with the same quantization matrix," *IEEE Transaction on Information Forensics and Security*, vol. 5, pp. 848–856, 2010.
- [5] Q. Liu, A.H. Sung, and M. Qiao, "A method to detect jpeg-based double compression," *Advances in Neural Networks ISNN Springer 2011*, vol. 6676, pp. 466–476, 2011.
- [6] G. Puglisi, A.R. Bruna, F. Galvan, and S. Battiato, "First jpeg quantization matrix estimation based on histogram analysis," in *Proc. IEEE International Conference on Image Processing*, September 2013, pp. 4502–4506.
- [7] Y. Zhang, S. Li, S. Wang, and Y.Q. Shi, "Revealing the traces of median filtering using high-order local ternary patterns," *IEEE Signal Processing Letters*, vol. 21, pp. 275–280, March 2014.
- [8] X. Kang, M.C. Stamm, A. Peng, and K.J.R. Liu, "Robust median filtering forensics using an autoregressive model," *IEEE Transactions on Information Forensic and Security*, vol. 8, pp. 1456–1468, September 2013.
- [9] M. Kirchner and R. Bohme, "Hiding traces of resampling in digital images," *IEEE Transactions on Information Forensic and Security*, vol. 3, pp. 582–592, November 2008.
- [10] C. Chen, J. Ni, and J. Huang, "Blind detection of median filtering in digital images: A difference domain based approach," *IEEE Transactions on Information Forensic and Security*, vol. 22, pp. 4699–4710, December 2013.
- [11] H. Zeng, T. Qin, X. Kang, and L. Liu, "Countering anti-forensics of median filtering," in *Proc. IEEE Int. Conf. on Acoustics Speech and Signal Processing*, May 2014, pp. 2704–2708.
- [12] V. Conotter, P. Comesana, and F. Perez-Gonzalez, "Joint detection of full-frame linear filtering and jpeg compression in digital images," in *Proc. IEEE Int. Workshop on Information Forensics and Security*, November 2013, pp. 156–161.
- [13] H. Cao and A.C. Kot, "Manipulation detection on image patches using fusionboost," *IEEE Transactions on Information Forensic and Security*, vol. 7, pp. 992–1002, June 2012.
- [14] M.A. Robertson and R.L. Stevenson, "Dct quantization in compressed images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 27–38, January 2005.
- [15] S.Z. Li and S. Singh, "Markov random field modeling in image analysis," *Springer*, vol. 3, 2009.
- [16] H. Ravi, A.V. Subramanyam, B. Avinash kumar, and G. Gupta, "Compression noise based video forgery detection," in *Proc. IEEE International Conference on Image Processing*, October 2014.
- [17] T. Pevny, P. Bas, and J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," *IEEE Transactions on Information Forensic and Security*, vol. 5, pp. 215–224, June 2010.
- [18] X. Zhao, S. Wang, S. Li, J. Li, and Q. Yuan, "Image splicing detection based on noncausal markov model," in *Proc. IEEE Int. Conf. on Image Processing*, September 2013, pp. 4462–4466.
- [19] D. Cozzolino, D. Gragnaniello, and L. Verdoliva, "Image forgery detection through residual-based local descriptors and block-matching," in *Proc. IEEE Int. Conf. on Image Processing*, October 2014, pp. 5297–5301.
- [20] G. Schaefer and M. Stich, "Ucid - an uncompressed colour image database," in *Proc. SPIE Storage and Retrieval Methods and Applications for Multimedia*, pp. 472–480.
- [21] Q. Liu, A. H. Sung, M. Qiao, Z. Chen, and B. Ribeiro, "An improved approach to steganalysis of jpeg images," *Information Sciences*, vol. 180, pp. 1643–1655, 2010, Software available at <http://www.shsu.edu/qxl005/New/Downloads>.
- [22] T. Gloe, "<http://forensics.inf.tu-dresden.de/ddimgdb/>."
- [23] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.