Forensic Analysis of Linear and Non Linear Image Filtering Using Quantization Noise

HAREESH RAVI, Indraprastha Institute of Information Technology, New Delhi, India
A. V. SUBRAMANYAM, Indraprastha Institute of Information Technology, New Delhi, India
SABU EMMANUEL, Kuwait University, Kuwait

The availability of intelligent image editing techniques and anti-forensic algorithms, make it convenient to manipulate an image and to hide the artifacts that it might have produced in the process. Real world forgeries are generally followed by the application of enhancement techniques such as filtering and/or conversion of the image format to suppress the forgery artifacts. Though several techniques evolved in the direction of detecting some of these manipulations, additional operations like re-compression, non linear filtering and other anti-forensic methods during forgery are not deeply investigated. Towards this, we propose a robust method to detect whether a given image has undergone filtering (linear or non linear) based enhancement, possibly followed by format conversion after forgery. In the proposed method, JPEG quantization noise is obtained using natural image prior and quantization noise models. Transition probability features extracted from the quantization noise are used for machine learning based detection and classification. We test the effectiveness of the algorithm in classifying the class of the filter applied and the efficacy in detecting filtering in low resolution images. Experiments are performed to compare the performance of the proposed technique with state-of-the-art forensic filtering detection algorithms. It is found that the proposed technique is superior in most of the cases. Also, experiments against popular anti-forensic algorithms show the counter anti-forensic robustness of the proposed technique.

CCS Concepts:•**Computing methodologies** → **Image manipulation;** *Computer graphics; Image processing;*

General Terms: Security, Verification

Additional Key Words and Phrases: Filtering Classification, Quantization Noise, Markov Process, Counter anti-forensics, HMRF, GMM

ACM Reference Format:
H. Ravi, A. V. Subramanyam, and S. Emmanuel 2015. Forensic Analysis of Linear and Non Linear Image Filtering Detection Using Quantization Noise. *ACM Trans. Multimedia Comput. Commun. Appl.* V, N, Article A (January YYYY), 23 pages.
DOI: http://dx.doi.org/10.1145/0000000.0000000

1. INTRODUCTION

Digital media can be easily acquired and modified using advanced software tools. The purpose of depicting the captured scenario through a multimedia would not be relevant if the content is modified with malicious intentions. Given the widespread usage of multimedia such as images and videos, and their gaining importance as evidence in a court of law, it is of paramount importance to know the manipulations an image has undergone. Even though there are active forensic techniques like [Bhatnagar et al. 2013], several passive techniques have been developed to detect whether a given image is authentic or tampered. Various types of forgery such as splicing and copy-move have

This work is supported by Department of Electronics and Information Technology (DeitY), Govt. of India, under the project "'Design and Development of Digital Multimedia Forgery Detection System with Project Number. 12(4)/2014-ESD.

DOI: http://dx.doi.org/10.1145/0000000.0000000

been investigated. Previous techniques like ([Farid 2009], [Chen et al. 2008], [Huang et al. 2010], [Liu et al. 2011] and [Puglisi et al. 2013]), consider JPEG double compression as a possible indication of forgery. This may not be true always, as an image may be simply decompressed and recompressed again. Since a forgery is often followed by an enhancement technique to make the forgery more convincing and less detectable, algorithms are recently being developed to detect enhancement operations in addition to the forgery. These operations include noise removal/addition, contrast enhancement, filtering enhancement, de-blurring, JPEG compression and edge enhancement etc. and can be performed after forgery [Conotter et al. 2013a]. These enhancements are also possible in encrypted domain in cloud storage devices [Lathey and Atrey 2015]. A typical forgery pipeline followed by an adversary is shown in the Fig 1. These enhancement operations can be considered as valid indications of forgery.

Out of these enhancement operations, filtering is one of the most commonly targeted technique in the forensic literature. Various techniques are proposed to detect linear filtering of images. For example, in [Conotter et al. 2013a], the authors proposed a Generalized Gaussian Distribution (GGD) based modeling of Discrete Cosine Transform (DCT) coefficients of a JPEG image for detecting whether the image is linearly filtered or not. Similarly, Conotter et al. in [Conotter et al. 2013b] proposed a set of histogram based statistical features to identify the type of linear filter applied to a JPEG image and its compression quality factor. Though the accuracy achieved by these techniques is high for linear filtering detection, filtering of TIFF images and robustness of the algorithm against double JPEG compression (i.e. when JPEG compressed image is filtered and re compressed) is not deeply investigated. This investigation is important as the artifacts of linear filtering enhancement that the image underwent prior to these operations might be suppressed [Kang et al. 2013].

Non-linear filtering, especially median filter detection has gained a lot of momentum lately as it can be used maliciously to hide fingerprints implanted during forgery. Authors in [Kang et al. 2013], [Zhang et al. 2014] and [Kirchner and Bohme 2008] are in consensus with the fact that, such non linear filtering operations can significantly reduce the performance of forensic techniques that assume linearity in detecting re-sampling or scaling of images. In order to counter this problem of reduced performance of forensic algorithms because of median filtering, techniques like [Zhang et al. 2014], [Chen et al. 2013], [Kirchner and Fridrich 2010] and [Zeng et al. 2014] were proposed to detect median filtering. In particular, [Zhang et al. 2014] uses higher order Local Ternary Patterns of an image as features to detect if a given image is median filtered or not. Whereas [Kang et al. 2013] fits an autoregressive model to median filter residual to determine if an image is median filtered or not. In [Chen et al. 2013], the authors introduced Global probability features of empirical Cumulative Distribution Function and Local Correlation features to perform blind median filter detection. M. Kirchner and J. Fridrich in [Kirchner and Fridrich 2010] propose features such as streaking artifacts and SPAM (Subtractive Pixel Adjacency Matrix) for the detection of median filtering and median filtering involving JPEG compression respectively.

However, in spite of obtaining a good performance by the algorithms mentioned above, the experiments do not deal explicitly with multiple format conversions such as JPEG to TIFF and TIFF to JPEG. Also, all the above techniques consider linear or non-linear filtering independently since they explicitly deal with the statistical differences caused by each type of filter. For example, authors in [Zhang et al. 2014] propose to detect only median filtering and not any other enhancement operation. Further, JPEG post processing of filtered images is evaluated only for quality factors 70 and 90 while JPEG double compression is not deeply investigated. The robustness of these algorithms against anti-forensic techniques is also not evaluated. Authors in [Stamm et al. 2013] give a review of various forensic techniques proposed in the last decade. It is shown that the algorithms that detect enhancement operations like median fil-

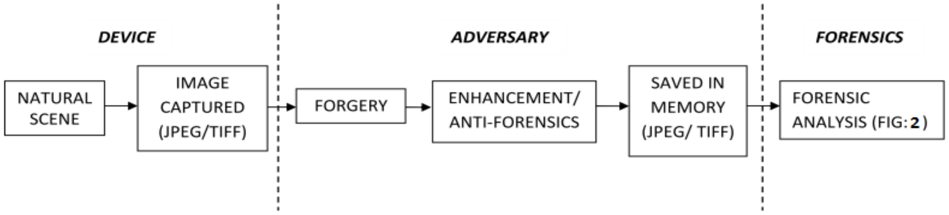
DEVICE
ADVERSARY
FORENSICS
NATURAL SCENE
IMAGE CAPTURED (JPEG/TIFF)
FORGERY
ENHANCEMENT/ ANTI-FORENSICS
SAVED IN MEMORY (JPEG/ TIFF)
FORENSIC ANALYSIS (FIG: 2)

Fig. 1. Block diagram of a typical forgery pipeline

tering or linear filtering are specific to the type of filter applied. i.e. filtering detection techniques in general deal with linear or non linear filtering independently.

Certain techniques like [Cao and Kot 2012] and [Qiu et al. 2014] consider multiple enhancement operations under some constraints. For example, authors in [Cao and Kot 2012] considered various kinds of manipulations including contrast enhancements in addition to linear and non linear filtering, obtaining a good accuracy using Fusion boost ensemble classifier. But, the data set considered is very small and the effect of format conversion and compression before and/or after the enhancement are not investigated. Similarly, in [Qiu et al. 2014], the authors experimented steganalysis models for enhancement detection considering filtering, contrast enhancement and compression. They evaluated popular steganalytic features and their application in forensics [Qiu et al. 2014]. However, compression was considered as a separate operation rather than a post/pre processing operation for other enhancements which is usually the case.

In order to overcome the aforementioned limitations, we propose an efficient technique[1] to detect images that are enhanced using operations such as Gaussian, Laplacian, average, sharpening or median filters as part of the forgery performed. The algorithm is observed to be able to classify the type of filter applied as low pass, high pass or median with a high accuracy. Also filtering detection performance is evaluated with low resolution images of dimension as low as 64×64. This helps in localization of the filtered part of an image [Kang et al. 2013]. In our experiments, image formats such as TIFF and JPEG are considered both before and after enhancement. Experiments involve filtered JPEG images that are saved as TIFF, filtered uncompressed TIFF images that are saved in JPEG format and filtered JPEG images double compressed while saving it in JPEG format again.

The block diagram of the proposed forensic technique is given in Fig 2. The technique is based on the principle that, when compression and filtering 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 and Stevenson 2005]. It can also be seen from Fig 3 that the average power spectral density of compression noise extracted from 300 unfiltered images follows low pass characteristics. We use the quantization noise model proposed in [Robertson and Stevenson 2005] that characterizes the spatial domain compression noise as a zero mean multivariate Gaussian distribution. We evaluate the performance of our algorithm using two different natural image models as proposed in [Fan et al. 2013] and [Li and Singh 2009] for quantization noise extraction. The noise thus extracted is modelled as a first order spatial ergodic Markov chain which has been proven to be an effective feature ([Chen et al. 2008], [Liu et al. 2011], [Ravi et al. 2014], [Pevny et al. 2010], [Fridrich and Kodovsky 2012]). These features are used to detect whether a given image has been filtered or not. In addition, we evaluate the performance of these features in classifying the type of filter applied and in detecting filtering in low resolution images for localization. The results are provided using standard UCID (Uncompressed Image Database) [Schaefer

[1] A preliminary version of this work is accepted in IEEE ICIP 2015 conference [Ravi et al. 2015].

and Stich 2004], NCID (Never Compressed Image Database) [Liu et al. 2010], Dresden [Gloe and Bohme 2010] and BOSS (Break Our Steganographic System) [Bas et al. 2011] image databases. Our contributions are as follows:

- We introduce a modified Huber Markov Random Field (HMRF) prior model which can incorporate the effect of blocking artifacts when an image undergoes compression. We also evaluate the impact of the modified prior on the detection accuracy.
- We propose quantization noise based transition probability features extracted from quantization noise to detect whether an image is filtered or not. Experimental setup comprises of a wide gamut of images of both uncompressed TIFF and, single and double compressed JPEG formats. The algorithm is shown to be effective in classifying the type of filter applied and also in detecting filtering in low resolution images for localization.
- The proposed method is compared with state-of-the-art filtering detection algorithms ([Conotter et al. 2013a] and [Zhang et al. 2014]) and is shown to perform better in most cases. Also, it is compared with popular feature extraction algorithms such as [Kirchner and Fridrich 2010], [Chen and Shi 2008], [Pevny and Fridrich 2007], [Kodovsky et al. 2012], [Cozzolino et al. 2014] and [Verdoliva et al. 2014] to show that the proposed technique gives better performance in filtering detection.
- Proposed technique's counter anti-forensic effectiveness is shown by testing it against state-of-the-art compression [Fan et al. 2013] and median filter [Fan et al. 2015] anti-forensic methods.

To the best of our knowledge, we believe this is the first image forensics algorithm which uses compression noise based transition probability features for image filtering detection, that targets various filters (linear and non-linear) for different image formats (JPEG and TIFF) [2] and forgeries. In addition, it is also robust against popular anti-forensic techniques. The notation adopted throughout this paper adheres to the following conventions.

- Scalar variables are denoted by lower case letters.
- Constants are denoted either by non-bold upper case letters or by greek symbols unless otherwise mentioned.
- Vectors are denoted by lower case bold letters.
- Matrices are denoted by upper case bold letters.
- $\mathbf{A}^T$ denotes transpose of matrix **A**.
- E[$\cdot$] denotes the expected value operator while **Q**[$\cdot$] denotes the quantization operator defined as $q \times round(\cdot/q)$ where q is the quantization factor.
- $p(a)$ denotes probability of a and $exp(a)$ denotes exponential of a.
- $\hat{\mathbf{A}}$ denotes estimate of **A**
- $\mathcal{S}$ denotes a closed set while $\mathcal{C}$ denotes a set of cliques. $\mathcal{N}(a|\mu, \sigma^2)$ denotes that a is normally distributed with mean μ and variance σ^2.
- a_i is a scalar that denotes i^{th} element of the vector a.
- $\mathbf{K}_p$ denotes p^{th} matrix of a set of matrices $\{\mathbf{K}_1,...,\mathbf{K}_{64}\}$ while $\mathbb{I}$ denotes the Identity matrix.

The rest of the paper is organized in the following way. Section 2 gives the related works that discusses the noise model and the prior models used while section 3 gives the modification of HMRF prior and then the actual proposed scheme of noise and feature extraction for both the priors. Section 4 contains the justification and evaluation of parameters and models used in our experiments. In Section 5 the experimental setup and results obtained for the proposed method for filtering detection are detailed. In Section 6, the advantages of this method over state of the art forensic algorithms

[2] JPEG is lossy compression, while TIFF is uncompressed.

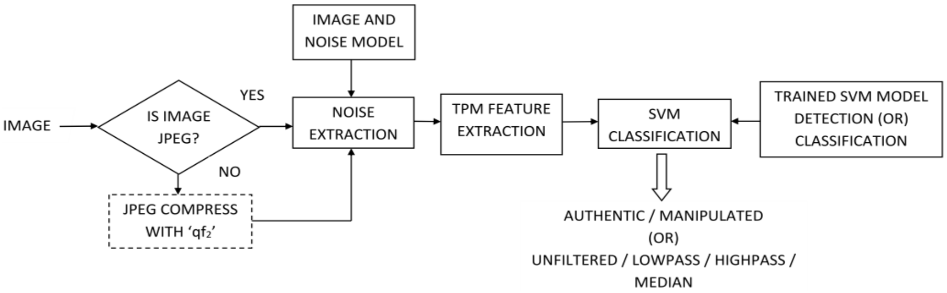

Fig. 2. Block diagram of the forensic analysis pipeline for authentication

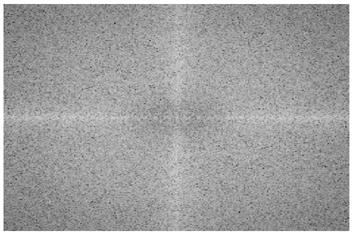

Fig. 3. Power Spectral Density of compression noise extracted from unfiltered images

and features are evaluated. It also gives the performance of the proposed technique in localizing filtering and detecting filtering when more than one filter is applied. In section 7 we discuss the Counter anti-forensic effectiveness of our method. Section 8 gives an outline comparison of the performance using the two image priors proposed while Section 9 concludes the paper.

2. PRELIMINARIES

Let $\mathbf{I}$ be a natural image scene in spatial domain of size $M \times N$. Let $\mathbf{z}$ be the vectorized form of a 8×8 block of the image $\mathbf{I}$ in spatial domain i.e. $\mathbf{z} = \{z_i\} \;\; \forall \;\; i \in [1, 2, \cdots, 64]$ where z_i represents each element of the vector, while $\mathbf{y}$ be the vectorized form of the corresponding block in DCT domain. Let $\mathbf{H}$ be the DCT matrix and $\mathbf{G}$ the Kronecker product of $\mathbf{H}$ with itself. Then the DCT operation during compression can be represented as,

$$\mathbf{y} = \mathbf{G}\mathbf{z} \Rightarrow \mathbf{z} = \mathbf{G}^{\mathbf{T}}\mathbf{y} \tag{1}$$

In case of lossy compression, the quantized image block in DCT domain is quantized as $\mathbf{y_q} = \mathbf{Q}[\mathbf{y}]$. The corresponding quantized image block in spatial domain for compression is given by $\mathbf{z_q} = \mathbf{G^T}\mathbf{y_q}$.

2.1. Quantization Noise Model

In lossy compression, some information is lost due to the rounding process that follows quantization. Quantization error in the spatial domain and frequency domain can be defined as the difference between the unquantized block and quantized block [Robertson and Stevenson 2005].

$$\mathbf{e_z} = \mathbf{z_q} - \mathbf{z} \text{ and } \mathbf{e_y} = \mathbf{y_q} - \mathbf{y} \tag{2}$$

The main parameter that is needed to model this quantization noise is the variance of individual frequency coefficients and the covariance matrix. The covariance matrix of error in the spatial domain can be represented as

$$\mathbf{K}_{\mathbf{e_z}} = E[(\mathbf{z_q} - \mathbf{z})(\mathbf{z_q} - \mathbf{z})^{\mathbf{T}}] = \mathbf{G}^{\mathbf{T}}\mathbf{K}_{\mathbf{e_y}}\mathbf{G} \tag{3}$$

where $\mathbf{K}_{\mathbf{e_y}} = E[(\mathbf{y_q} - \mathbf{y})(\mathbf{y_q} - \mathbf{y})^{\mathbf{T}}]$. Thus, the spatial domain quantization error can be represented as,

$$p(\mathbf{e_z}) = \frac{1}{(2\pi)^{D/2}|\mathbf{K}_{\mathbf{e_z}}|^{1/2}} exp\bigg(- \frac{1}{2}\mathbf{e_z^T}\mathbf{K}_{\mathbf{e_z}}{}^{-1}\mathbf{e_z}\bigg) \tag{4}$$

where $D = 64$ is the number of dimensions of the multivariate Gaussian. The probability distribution derived in eq (4) gives the quantization noise model. This zero mean multivariate distribution models the quantization noise for each non overlapping block of an image. Since JPEG compression is a block wise operation, we assume that the quantization noise of all the blocks are independent to each other.

2.2. Markov Random Field Prior

Markov Random Field (MRF) has been widely used to model the statistical properties of natural images [Li 1995]. The conditional distribution for any pixel $v = (i, j)$ in an image $\mathbf{I}$ can be written as

$$p(\mathbf{I}_v|\mathbf{I}_{\mathcal{C}-v}) = p(\mathbf{I}_v|\mathcal{N}_v) \tag{5}$$

where $\mathcal{N}_v$ is the local 8-connected neighborhood at v and $\mathcal{C} \in \mathcal{N}_v$ is a set of sites in the image. Using the MRF, we can compute the joint probability distribution of a block $\mathbf{z}$ of a natural image given by Gibbs measure as,

$$p(\mathbf{z}) = \frac{1}{\beta} exp\bigg(- \lambda_H \sum_{c\in\mathcal{C}} U(\mathbf{z}_c)\bigg) \tag{6}$$

where, β is a normalization constant, $U(\cdot)$ is the energy function and λ_H is a free parameter. These parameters will be explained in section 3.1.

2.3. Gaussian Mixture Prior

Learning based image priors have been introduced as good priors for image restoration tasks [Zoran and Weiss 2011]. In the proposed method, we use a Gaussian Mixture Model (GMM) based prior from [Fan et al. 2013]. The probability distribution that models a natural image patch $\mathbf{z}$ of an image $\mathbf{I}$ is given as,

$$p(\mathbf{z}) = \sum_{\nu=1}^{M_c} \pi_\nu \mathcal{N}(\mathbf{z}|\mu_\nu, \mathbf{\Sigma}_\nu) \tag{7}$$

where π_ν are the mixing weights for each mixture component ν, M_c is the total number of mixture components and μ_ν and $\mathbf{\Sigma}_\nu$ are the corresponding mean and covariance matrix respectively. The means, covariance matrices and mixing weights are learned using Expectation Maximization (EM) algorithm as explained in section 3.3.

3. PROPOSED SCHEME

Given an image, the quantization noise model and the image prior models described in section 2 are used to extract quantization noise. The extracted noise is then modelled as a first order spatial Markov chain to extract transition probability features. The transition probability features are used to detect if a given image is manipulated or not. We first explain the modified HMRF image prior model.

3.1. Modified Huber Markov Random Field Prior

We propose a modified Huber Markov Random Field model derived from the joint distribution derived in eq (6). The energy function in the distribution is given by Huber function as,

$$p(\mathbf{z}) = \frac{1}{\beta} exp\Big(- \lambda_H \sum_{c \in \mathcal{C}} \rho_T(\mathbf{d_c^t z})\Big) \tag{8}$$

where, $\mathbf{d_c^t}$ extracts the difference between a pixel and its 8 immediate neighbours in $\mathbf{z}$ as given later in eq (10), $\rho_T(\cdot)$ is the Huber function defined for clique $c \in \mathcal{C}$ as,

$$\begin{aligned} \rho_T(u) &= \begin{cases} wu^2, & |u| <= T, \\ w(T^2 + 2T(|u| - T)), & |u| > T \end{cases} \\ w &= \begin{cases} 1 & \forall\ z_u : u \notin \mathcal{S}, \\ \gamma & \text{otherwise} \end{cases} \end{aligned} \tag{9}$$

where w is introduced as a weight parameter to incorporate the effect of compression on an image and T is a threshold. The parameters λ_H, γ and T are empirically determined as given in section 5. $\mathcal{S}$ is the set of pixels which belong to the border pixels in each 8×8 block. The prior model degenerates to,

$$p(\mathbf{z}) = \frac{1}{\beta} exp\Big(- \lambda_H \sum_{m=0}^{M_b - 1} \sum_{n \in N_m} \rho_T(z_n - z_m)\Big) \tag{10}$$

Where N_m is the index set of neighbors for the m^{th} pixel, and M_b is the number of pixels in the block. The probability distribution described above gives the modified HMRF prior model of a natural image scene.

3.2. Noise extraction using modified-HMRF image prior

We use Bayesian maximum *a posteriori* (MAP) estimation for extracting the compression noise using modified HMRF prior model. The MAP criterion is,

$$\hat{\mathbf{z}} = \arg\max_{\mathbf{z}} p(\mathbf{z}|\mathbf{z_q}) = \arg\max_{\mathbf{z}} p(\mathbf{z})p(\mathbf{z_q}|\mathbf{z}) \tag{11}$$

where $\hat{\mathbf{z}}$ is the final estimate for the block after removing the compression noise. The $p(\mathbf{z_q}|\mathbf{z})$ term in the above equation is a Gaussian random variable with mean $\mathbf{z}$ and auto covariance $\mathbf{K_{e_z}}$. The covariance matrix $\mathbf{K_{e_z}}$ of the noise model is derived in the following manner. Let an individual DCT coefficient be p and the quantization step size be q, then we can write, $p_q = round(p/q)$. Assuming that p_q is distributed uniformly over the interval $[p - q/2, p + q/2]$ [Robertson and Stevenson 2005], the variance of the quantization error would be $q^2/12$. Since the DCT coefficients are independent, the DCT domain covariance matrix will be a diagonal matrix,

$$\mathbf{K_{e_y}} = \begin{bmatrix} \frac{{q_1}^2}{12} & 0 & \cdot\ \cdot & 0 \\ 0 & \frac{{q_2}^2}{12} & \cdot\ \cdot & 0 \\ \cdot & \cdot & \cdot\ \cdot & \cdot \\ \cdot & \cdot & \cdot\ \cdot & \cdot \\ 0 & 0 & \cdot\ \cdot & \frac{{q_{64}}^2}{12} \end{bmatrix} \tag{12}$$

where $\{q_1, q_2, ..., q_{64}\}$ are derived from the quantization table. Covariance matrix $\mathbf{K_{e_z}}$ is obtained from eq (3) using the above derived DCT domain covariance matrix $\mathbf{K_{e_y}}$.

Now eq (11) can be written as

$$\hat{\mathbf{z}} = \arg\max_{\mathbf{z}} \Bigg\{ \frac{1}{\beta} exp\Big(-\lambda_H \sum_{m=0}^{M_b-1} \sum_{n\in N_m} \rho_T(z_n - z_m) \Big) \Big(\frac{1}{(2\pi)^{D/2}|\mathbf{K}_{\mathbf{e_z}}|^{1/2}} exp\Big(-\frac{1}{2}\mathbf{e}_{\mathbf{z}}^T \mathbf{K}_{\mathbf{e_z}}{}^{-1}\mathbf{e_z} \Big) \Big) \Bigg\} \tag{13}$$

In order to maximize eq (11), the arguments of $exp(.)$ in eq (13) is minimized using gradient descent algorithm. Denoting the estimate of $\mathbf{z}$ at iteration t as $\mathbf{z}^{(t)}$, the gradient descent update for the next iteration would be,

$$\mathbf{z}^{(t+1)} = \mathbf{z}^{(t)} - \zeta^{(t)}\boldsymbol{\psi}(\mathbf{z}^{(t)}) \tag{14}$$

where, $\zeta^{(t)}$ is a step size that ideally reduces the objective function as much as possible. The gradient $\psi(\mathbf{z})$ of the objective function is given as,

$$\boldsymbol{\psi}(\mathbf{z}) = \lambda\nabla\boldsymbol{\phi}_1(\mathbf{z}) + \nabla\boldsymbol{\phi}_2(\mathbf{z}) \tag{15}$$

where the individual terms are obtained from the eq (13) as,

$$\begin{aligned} \boldsymbol{\phi}_1(\mathbf{z}) &= \sum_{c\in C}\rho_T(\mathbf{d}_{\mathbf{c}}^{\mathbf{t}}\mathbf{z}) \\ \boldsymbol{\phi}_2(\mathbf{z}) &= \frac{1}{2}(\mathbf{z}-\mathbf{z_q})^T\mathbf{K}_{\mathbf{e_z}}{}^{-1}(\mathbf{z}-\mathbf{z_q}) \end{aligned} \tag{16}$$

We use a constantly decreasing step size ζ starting from 0.1 and reduced at every iteration by a factor of $0.05 \times \zeta$. The maximum number of iterations is set at 500 and tolerance level for the change in cost is set at 0.0001. The algorithm terminates if the tolerance level or maximum iteration is reached. λ in the above equation determines the amount of smoothness and is empirically determined [Robertson and Stevenson 2005]. Combining all the resulting denoised blocks $\hat{\mathbf{z}}$, we generate the denoised image $\hat{\mathbf{I}}$. The compression noise $\mathbf{N_c}$ is obtained as $\mathbf{N_c} = \mathbf{I} - \hat{\mathbf{I}}$.

3.3. Noise extraction using GMM image prior

We use the Expected Patch Log Likelihood (EPLL) method introduced in [Zoran and Weiss 2011] where prior information about the image to be restored is used for restoration as

$$EPLL_p(\mathbf{I_q}) = \sum_i \log p(\boldsymbol{\Omega}_i\mathbf{I_q}) \tag{17}$$

where $\boldsymbol{\Omega}_i$ is the matrix that extracts the $\mathbf{i}^{th}$ patch from the image $\mathbf{I_q}$ out of all the *overlapping patches*, while $\log_p(\boldsymbol{\Omega}_i I_q)$ is likelihood that the $\mathbf{i}^{th}$ patch is under the prior p and λ_G a regularization parameter. We divide all the overlapping patches to 64 types/sets of patches $\mathcal{S}_p \in \{\mathcal{S}_1, \mathcal{S}_2, ..., \mathcal{S}_{64}\}$, according to their relative position with respect to JPEG blocks ([Fan et al. 2013]). Also, the prior $p(\boldsymbol{\Omega}_i\mathbf{z})$ is replaced with the GMM prior introduced in section 2.3 and if at any point, a particular patch is represented as z, the optimization problem using eq (17) and compression noise model from eq (4) will become

$$\hat{\mathbf{I}} = \arg\min_{\mathbf{z}} \left\{ \frac{\lambda_G}{2}\sum_{p=1}^{64}\sum_{\boldsymbol{\Omega}_i\in\mathcal{S}_p}(\boldsymbol{\Omega}_i\mathbf{e_z})^T\mathbf{K}_p^{-1}(\boldsymbol{\Omega}_i\mathbf{e_z}) - \sum_i \log p(\boldsymbol{\Omega}_i\mathbf{z}) \right\} \tag{18}$$

where $\mathbf{\Omega}_i$ is a matrix extracting the i-th patch from the noise $\mathbf{e_z}$, and $\mathbf{K}_p$ is the covariance matrix for modeling the quantization noise of the p-th group of patches. $\mathbf{K}_p$ is obtained by training real quantization noise in the following way. 300 random images from the UCID [Schaefer and Stich 2004] database are compressed with quality factor of the image under consideration and subtracted from the respective original images to obtain the actual quantization noise. This noise is used to extract 6016 8×8 patches from each image (94 patches of each of the 64 types, $300 \times 94 \times 64$ patches totally). These noise patches are then used to get the 64 covariance matrices $\mathbf{K}_p$ corresponding to the types of patches. The term $\log p(\mathbf{\Omega}_i\mathbf{z})$ gives the expected patch log likelihood i.e.

$$\log p(\mathbf{\Omega}_i\mathbf{z}) = \log\left(\sum_{\nu=1}^{M_c} \pi_\nu \mathcal{N}(\mathbf{\Omega}_i\mathbf{z}|\mu_\nu, \mathbf{\Sigma}_\nu)\right) \tag{19}$$

The prior model $p(\mathbf{\Omega}_i\mathbf{z})$ is obtained by training 338×6000 overlapping patches of natural images from the UCID database. EM algorithm is used to train 200 mixture components according to eq (7). The denoised patch corresponding to the noisy patch is obtained by an approximate MAP procedure [Zoran and Weiss 2011] using Wiener filter solution for ${\nu_{\max}}^{th}$mixture component as

$$\hat{\mathbf{z_p}} = (\mathbf{\Sigma}_{\nu_{\max}} + \mathbf{K}_p^2\mathbb{I})^{-1}(\mathbf{\Sigma}_{\nu_{\max}}\mathbf{z_q} + \mathbf{K}_p^2\mathbb{I}\mu_{\nu_{\max}}) \tag{20}$$

where $\nu_{\max} = \arg\max_\nu \pi_\nu$ is the mixture component that has the highest conditional mixing weight, i.e. $\pi_\nu = p(\nu|\mathbf{z_q})$ and $\mathbb{I}$ is the identity matrix. Then $\mathbf{\Sigma}_{\nu_{\max}}$ will be the covariance matrix of the ${\nu_{\max}}^{th}$ component of the prior model whereas $\mathbf{K}_p$ is the noise covariance matrix trained for the p-th type of patch $\mathcal{S}_p$. An estimate $\hat{\mathbf{z_p}}$ for each type of patch is obtained by the MAP approximation and then combined as given in eq (18) to get the image estimate $\hat{\mathbf{I}}$. The compression noise $\mathbf{N_c}$ is obtained as $\mathbf{N_c} = \mathbf{I} - \hat{\mathbf{I}}$.

3.4. 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, $p(X_{t+1} = x|X_1 = x_1, X_2 = x_2, ..., X_t = x_t) = p(X_{t+1} = x|X_t = x_t)$, where X_{t+1} is the present state and $(X_1, X_2, ..., X_t)$ are the previous states. The features that we use to characterize this noise is Transition Probability Matrix (TPM). TPM characterizes a Markov chain by providing the probability of transition between each state. This is extracted by modeling the elements of the difference array (gradient along eight directions) as states. It was found experimentally that eight directions rather than just four, gave a better result.

Difference arrays are obtained from the noise matrix as shown in eq (21). Representation along the right (East →) direction is shown hereafter and those along other directions (West ← , North ↑ , South ↓ , North East ↗ , North West ↖ , South East ↘ , South West ↙) can be obtained in a similar way.

$$\mathbf{D}_{\mathbf{c}}^{\rightarrow}(i,j) = \mathbf{N_c}(i,j) - \mathbf{N_c}(i,j+1) \tag{21}$$

where $i = 1, 2..., M$ and $j = 1, 2..., (N-1)$ are indices representing each element in the matrix. Values in each difference array $\mathbf{D}_{\mathbf{c}}^{\rightarrow}$ are rounded off to the nearest integer to get integer value states as given in eq (22) where q = 1. It is then truncated between $-T_r$ to T_r as shown in eq (23) before extracting the transition probabilities.

$$\tilde{\mathbf{D}}_{\mathbf{c}}^{\rightarrow}(i,j) = round(\tilde{\mathbf{D}}_{\mathbf{c}}^{\rightarrow}(i,j)/q) \tag{22}$$

$$\tilde{\mathbf{D}}_{\mathbf{c}}^{\rightarrow}(i,j) = \begin{cases} -T_r, \mathbf{D}_{\mathbf{c}}^{\rightarrow}(i,j) < -T_r \\ +T_r, \mathbf{D}_{\mathbf{c}}^{\rightarrow}(i,j) > +T_r \\ \mathbf{D}_{\mathbf{c}}^{\rightarrow}(i,j), otherwise \end{cases} \tag{23}$$

This provides us with $(2T_r + 1)$ different states to model the Markov chain. Now, TPM is constructed in each direction from each difference array as follows,

$$p_{u_1,u_2}^{\rightarrow} = p(\tilde{\mathbf{D}}_{\mathbf{c}}^{\rightarrow}(i,j+1) = u_1 | \tilde{\mathbf{D}}_{\mathbf{c}}^{\rightarrow}(i,j) = u_2) \tag{24}$$

where, $u_1, u_2 \in [-T_r, T_r]$, and $u_1, u_2 \in \mathbb{Z}$. Similarly, the probabilities can be obtained for other directions. This gives us $(2T_r + 1) \times (2T_r + 1)$ transition probabilities for each difference array. The TPMs along the eight directions are concatenated to get the final feature vector $(2T_r + 1) \times (2T_r + 1) \times 8$ and this remains the same irrespective of the size of the image.

4. JUSTIFICATION OF PARAMETERS AND MODELS

First we define values for the parameters used in our models and experiments with plausible explanation for the same. We then justify the usage of compression noise as a metric to detect the presence of filtering.

The Huber Markov Random Field model introduced in the section 2.2 is modified to introduce the weight parameter w. Experiments are performed with different weights and w is chosen to be 5 for high accuracy. Also, λ_H in eq (13) is set to 0.1 while γ and T in eq (9) are set to 5 and 10 respectively. These are empirically determined to give the best results. In our model T_r in eq (23) is set to 15 for HMRF prior based experiments and to 8 for GMM prior based experiments as 95% of the values in the difference array of noise are in [-15, 15] and [-8, 8] respectively. This gives a 7688 [3] and 2312 dimensional feature vectors for HMRF and GMM prior based techniques. Also, experiments were performed with values between 3 to 30 for T_r and it was found that saturation in performance reached after 15 for HMRF and 8 for GMM. Moreover, for values lower than the above mentioned T_r, the accuracy in performance drops significantly.

In order to analyze the difference between the TPM features of filtered and unfiltered images, we give a justification by modeling low pass filtering as a linear process. In the following, we analyze the effect of filtering for GMM prior model and a similar analysis can be done for HMRF model or for other filters. He et al. proposed a linear filtering model in [He et al. 2013] as given in eq (25).

$$z_{f_i} = a_k I_i + b_k, \forall i \in C_k \tag{25}$$

where i is the index of a pixel, k is the index of a local square window C_k and z_{f_i} is output pixel [He et al. 2013]. Here, a_k and b_k are a function of local neighborhood C_k of the input image p and guidance image I. Similarly, the linear operation that denotes low pass filtering can be denoted as,

$$z_{f_i} = a_i \times z_i + b_i \quad \forall i \in C \tag{26}$$

where we have removed the dependency of the window on k as the filter is spatially invariant. Here, z_i is the pixel being modified. a_i is a function of local neighborhood C of the input image. Since the local neighborhood is dynamic in nature, a_i can be modeled as a random variable. Due to round-off of z_{f_i} to nearest integer, we introduce b_i which can model the noise incurred in the rounding-off. In order to make the computation tractable, we assume that a_i and b_i are independent Gaussian random variables with

[3] It is to be noted the low dimensional linear projection of the feature set using PCA or by averaging reduced the efficiency of the algorithm. Hence the entire feature set it used for classification.

mean μ_a and 0 respectively, and, variance σ_a^2 and σ_b^2 respectively. In eq (26), the first term is a product of two independent Gaussian random variables. The pdf of this term is given by [Ware and Lad 2003]

$$\int_{-\infty}^{\infty} \frac{1}{|a|} \sum_{k=1}^{K} \pi_k \mathcal{N}(\frac{z}{a}|\mu_k, \mathbf{\Sigma}_k) \frac{e^{(-\frac{(a-\mu_a)^2}{2\sigma_a^2})}}{\sqrt{2\pi\sigma_a^2}} da \tag{27}$$

Towards this, we analyze the pdf of two independent Gaussian random variables and eq (27) can then be treated as sum of product of independent Gaussian random variables, z_i and a_i. However, a closed form expression for eq (27) would be very difficult to compute. Therefore, we analyze the Moment Generating Function (MGF) in order to check if the pdf of z_f can still be GMM. In [Ware and Lad 2003], it is shown that the MGF of product of two independent Gaussian random variables is,

$$M_{z_f}(t) \rightarrow (1 - \sigma_a^2 \mathbf{\Sigma}_k t^2)^{-1/2} \tag{28}$$

This means the MGF of z_f does not follow a Normal distribution and consequently it is not a GMM distribution. Since, we assume a GMM prior model for images, the distribution of filtered images is different than that of unfiltered images. It would be reasonable to assume that the compression noise and TPM generated from noise of these images would also be different in both the cases which is supported by the experimental results. Also, the transition probability features used in the experiments are known to be effective in capturing neighborhood information from previous works ([Ravi et al. 2014], [Pevny et al. 2010]). In Fig 4, the average of histograms of TPM features extracted from unfiltered, low pass, high pass and median filtered images are given. Distinction in the TPM histogram can be seen to be significant enough to support the experimental results. This distinction is visible when the experiments are performed using either of the image priors (refer Table III). It can be observed from Fig 5 that the power spectral density of compression noise extracted from filtered images show characteristics of the applied filter. This demonstrates that compression noise is a good measure for detection and classification of the applied filter.

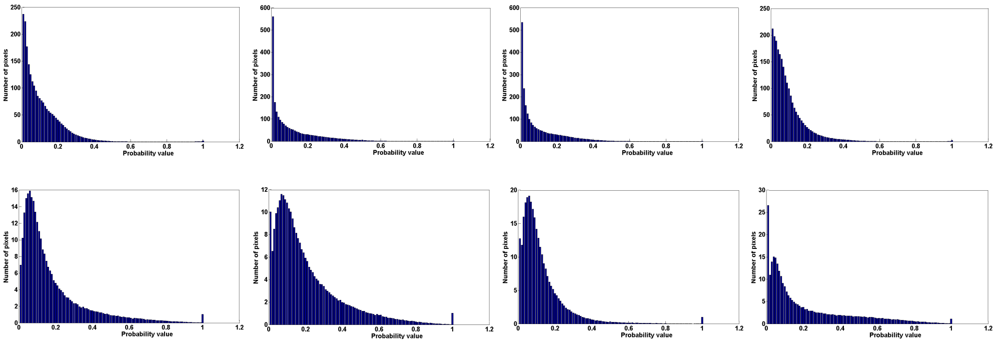

Fig. 4. Average histogram of TPM extracted from compression noise using Top row: GMM prior of unfiltered, low pass, median and high pass (left to right) filtered images Bottom row: HMRF prior of unfiltered, low pass, median and high pass (left to right) filtered images.

5. EXPERIMENTAL SETUP AND RESULTS

We consider 1338 uncompressed images from the UCID database [Schaefer and Stich 2004] along with 1262 random Never compressed images of 5150 images of the NCID database [Liu et al. 2010]. All the 1338 UCID images are cropped to 256×256 from the

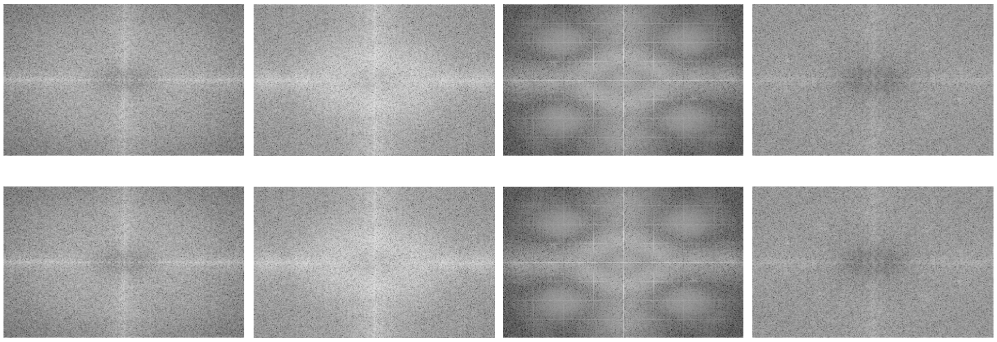

Fig. 5. Average Power spectral density estimate of compression noise using Top row: GMM prior from unfiltered, low pass, median and high pass (left to right) filtered images Bottom row: HMRF prior from unfiltered, low pass, median and high pass (left to right) filtered images.

center, matching the size of NCID images (Though other dimensions can be used as it will give the same dimensional feature, we use this to save time in processing). We also take 400 single compressed images captured using different digital cameras from the Dresden image database [Gloe and Bohme 2010]. This together makes 3000 authentic images of size 256×256 called the 'original set' (2600 uncompressed TIFF images and 400 single compressed JPEG images). Another 1000 random NCID images not present in the previous 1262 are single compressed with a randomly chosen factor $qf_1 \in (30, 90]$, $qf_1 \in \mathbb{Z}$. This set is called the 'splicing set'. We generate two classes of images for our experiments from these sets as given in Table I. In the experiments, a JPEG/TIFF image is forged, enhanced and then saved in either JPEG or TIFF format following a typical forgery pipeline. When the given final image is in uncompressed TIFF format, it is JPEG compressed, referred in the Table I as 'Forensic end'. The parameters in the table are to be read as follows, $qf_1 \in (30, 90]$, $qf_2 \in \{30, 40, 50, 60, 70, 80, 90\}$ as in the Table III, 'copy-move' forgery is of size $s \times s$ where $s \in (50, 130]$, $s \in \mathbb{Z}$ and 'splicing' forgery is 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 3 (for both HMRF and GMM priors separately) from 6000 (3000 unfiltered and 3000 filtered) images in total for each quality factor qf_2 as in the Table III.

Table I. Various image manipulation pipelines considered in experiments - To be read from left to right

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)	copy-move / splicing	filtering from Table II	TIFF/JPEG-qf_2	JPEG (qf_2)

Table II. Enhancement techniques performed as part of forgery

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

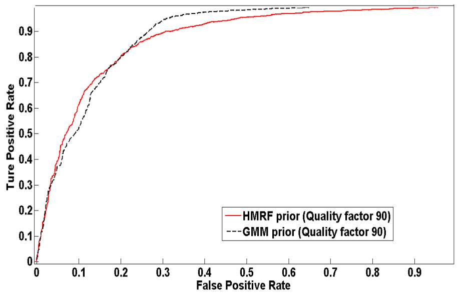

Fig. 6. ROC curve for the proposed method

For further experiments we define the following terms. 'Image manipulation pipeline' is defined as the pipeline of processes that an image goes through during manipulation. 'Forensic compression' means that, compression is performed as part of the proposed algorithm and not as part of the manipulation.

5.1. Filtering Detection

The first set of experiments are carried out to detect filtering in an image. 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 use Radial Basis Function (RBF) kernel based binary classification Support Vector Machine (SVM) from [Chang and Lin 2011] libsvm library. Grid search is performed for determining the parameters that give the best average cross validation [5] accuracy which is provided as $(TPR+TNR)/2$ in Table III where TPR is the True Positive Rate and TNR is the True Negative Rate. It can be observed from the table that the classification accuracy between an unfiltered and a filtered image is above 80% for most cases in both the prior models. In Fig 6 the ROC curve of classification using HMRF prior and GMM prior for quality factor 90 is given. High TPR is achieved for small FPR in spite of the various 'real world situations' replicated in the experiments. It can be seen that both HMRF prior and GMM prior based experiments give similar performance.

Table III. Detection accuracy for various quality factors (qf_2)

qf_2	HMRF PRIOR (%)	GMM PRIOR (%)
30	80.45	78.65
40	82.51	79.20
50	78.55	80.00
60	81.85	80.60
70	80.93	81.00
80	84.75	82.06
90	82.50	83.50

[4] Values are parameters used in MATLAB for specific filters.

[5] All the accuracy values reported in this paper are computed as the average of 5 experiments using different random combinations of training and testing data for the best cost and gamma value determined by grid search for RBF kernel.

5.2. Filter classification

The next set of experiments are performed to identify the type of filter applied. In order to do this, we generate four classes of images namely unfiltered, median filtered, low pass filtered and high pass filtered. We generate 650 images for each class from 2600/3000 total images of the 'original set'. Now, SVM is trained for one vs all multi class classification using 50% of samples from each class. Detection accuracy for each class using noise extracted from GMM prior and HMRF prior are given in the Table IV. The image manipulation pipeline used in this set of experiments is modified from [Conotter et al. 2013a] as JPEG $(qf_1)\xrightarrow{filtering}$TIFF to TIFF/JPEG $(qf_1)\xrightarrow{filtering}$TIFF incorporating TIFF images in the source too. Since the final image is TIFF, it is JPEG compressed at the forensic end with quality factor 90. The accuracy is given in Table IV for different qf_1 like given in [Conotter et al. 2013a]. It can be seen from the table that the accuracy in positively identifying a median filter or an unfiltered image is high using both the priors. Also, HMRF prior based experiments give better accuracy while detecting low pass filtering than high pass while GMM prior classifies high pass filtered images better than low pass filtered images.

Table IV. Classification accuracy for various quality factors (qf_1)

QF (qf_1)	ACCURACY FOR HMRF PRIOR (%)				ACCURACY FOR GMM PRIOR (%)			
	Unfiltered	Low pass	High pass	Median	Unfiltered	Low pass	High pass	Median
30	88.92	84.31	82.46	94.77	93.23	80.00	88.62	95.08
40	88.92	85.54	79.08	93.85	90.15	82.15	88.00	95.69
50	84.92	85.85	79.69	94.15	89.23	85.23	84.62	93.85
60	87.08	82.15	77.54	95.38	89.85	82.46	86.77	95.08
70	86.15	87.08	80.62	94.77	90.77	83.08	85.85	96.31
80	88.62	84.62	83.08	95.69	90.77	82.46	86.77	96.62
90	89.54	83.08	85.54	96.62	89.54	82.87	88.00	96.00

6. PERFORMANCE EVALUATION AND COMPARISON

To evaluate the performance of the proposed technique we compare it with state of the art filtering detection algorithms and popular feature extraction algorithms. Also, the efficiency of this algorithm in detecting filtering on low resolution images is discussed. The details of the experiments performed and the detection accuracy achieved are given below.

6.1. Comparison with filtering detection algorithms

We compare the efficiency of the proposed technique for image manipulation pipelines considered in [Conotter et al. 2013a] and [Zhang et al. 2014]. The database of filtered and unfiltered images are created using the image formats and filters mentioned in these papers and the proposed algorithm is applied over the created database.

6.1.1. Linear filtering detection. We perform an experiment with 2000 images from the 'original set' to compare our results with state of the art [Conotter et al. 2013a] linear filtering detection technique. The 'Filtered class' of 1000 images is obtained by following the manipulation pipeline, JPEG $(qf_1)\xrightarrow{filtering}$TIFF as in [Conotter et al. 2013a]. To implement our algorithm for this pipeline, forensic compression of the final TIFF image is done with JPEG quality factor 90. Another 1000 images from the 'Unfiltered

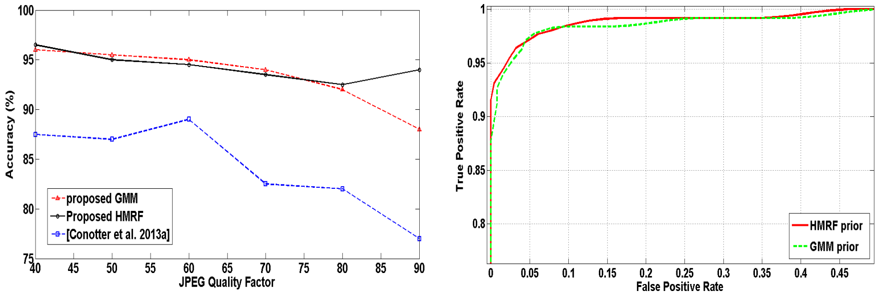

Fig. 7. *Left*: Detection accuracy comparison for various quality factors (qf_1) of JPEG compression, *Right*: ROC curve using the proposed method on image manipulation pipeline (refer section 6.1.1) as given in [Conotter et al. 2013a] for qf_1 = 30.

class' as obtained for the original experiments for quality factor 90 constitute the unfiltered class for this experiment. Classification is done using 50% (500 authentic, 500 manipulated) samples for training and the rest for testing. The accuracy achieved in this experiment using the proposed algorithm using both HMRF and GMM priors and that of [Conotter et al. 2013a] for each qf_1 is given in Fig 7. Accuracy is found to be higher for all qf_1 when compared with [Conotter et al. 2013a]. The ROC curve for linear filtering detection using the proposed method is given in Fig 7. It is seen from the Fig 7 that very high TPR of above 0.9 is achieved for a FPR of 0.05 when the proposed method is implemented on experimental pipeline considered in [Conotter et al. 2013a] for $qf_1 = 30$. This also is the case with both the image priors.

6.1.2. Median filter detection. Another experiment to detect only 'median filtering' is performed using 2000 images. The 'Filtered class' here is obtained using the manipulation pipeline TIFF$\xrightarrow{median filter}$JPEG ($qf_2$) as given in [Zhang et al. 2014] where $qf_2 = \{70, 90\}$. Experimental results are given only for $qf_2 = \{70, 90\}$ as results for only those quality factors are available in [Zhang et al. 2014]. 1000 images constitute the 'Filtered class' for each qf_2 and kernel size of the median filter.'Unfiltered class' for the corresponding qf_2 contains 1000 images from the 'Unfiltered class' of the original experiments. Detection accuracy is given in Table V for each qf_2 and median filter kernel using the proposed method and that of [Zhang et al. 2014]. The proposed method gives better or comparable performance in all the cases. In Fig 8, ROC curve for classification of 3×3 median filtering of JPEG images compressed with quality factor 90 using proposed method is given. The TPR is 0.97 for a very low FPR of 0.02 proving the efficacy of the proposed technique for median filter detection.

Table V. Detection accuracy for median filtering using [Zhang et al. 2014] and proposed technique

MEDIAN FILTER SIZE	3×3		5×5	
QUALITY FACTOR (qf_2)	90	70	90	70
[Zhang et al. 2014] (%)	98	94.5	**98.5**	**97.5**
Proposed HMRF prior (%)	**99**	**95.25**	**98.5**	96.5
Proposed GMM prior (%)	**98**	88	97.8	96

6.2. Comparison with popular features

We evaluate and compare the proposed algorithm with popular feature extraction algorithms like Chen et al.'s inter and intra block markov features [Chen and Shi 2008],

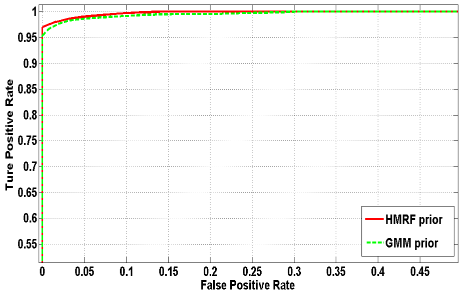

Fig. 8. ROC curve using the proposed method on image manipulation pipeline (refer section 6.1.2) as given in [Zhang et al. 2014] for qf_2 = 90.

T. Pevny and J. Fridrich's features as described in [Pevny and Fridrich 2007], Kodovsky et al.'s features on compact rich models for steganalysis as given in [Kodovsky et al. 2012], SPAM [Kirchner and Fridrich 2010] features, residual based local descriptors given in [Cozzolino et al. 2014] and camera-based co-occurrence features as proposed in [Verdoliva et al. 2014]. These features are named as proposed GMM, Chen, CC-Pev, CF, SPAM, ResF and CamF respectively for usage in this paper. Since SPAM and CamF features have considerably lesser dimensionality compared to the proposed GMM, we also perform experiments after increasing the dimension of both SPAM and CamF features to 2662 and 3281 obtained by using threshold values of 5 and 4 respectively. It is to be noted that this number is higher than the proposed GMM and these extended features are named as SPAMext and CamFext. We use the BOSS database consisting of 10000, 512×512 images for this experiment to show that the proposed GMM features do not overfit the data. The high cardinality of the dataset is used specifically to show that the proposed algorithm is not affected by large number of samples. Since CamF features are camera based, we use UCID database for the evaluation of these features as the entire UCID database is from the same camera. Also we resort to ROC curves for comparison with this algorithm as mentioned in the paper [Cozzolino et al. 2014]. We evaluate the general filtering detection performance of these features along with the proposed method on a dataset of 5000 unfiltered and 5000 filtered 256×256 images created from the BOSS database.

Table VI. Detection accuracy for various quality factors using the proposed scheme and popular features described in section 6.2

Features $\backslash QF$	30	60	90
Proposed GMM	**86.5**	**89.2**	**93.4**
SPAM [Kirchner and Fridrich 2010]	78	82.6	90.4
Chen [Chen and Shi 2008]	77.8	83.4	89.5
CC-Pev [Pevny and Fridrich 2007]	79.4	84.2	90.2
CF [Kodovsky et al. 2012]	82.2	85.4	90.4
ResF [Cozzolino et al. 2014]	82.14	83.45	85.56
SPAMext [Kirchner and Fridrich 2010]	80.6	83.43	91.2

It can be interpreted from Table VI that the proposed method gives better performance than popular features in all cases. For CamF and CamFext features, the ROC curve is plotted as mentioned in the paper [Cozzolino et al. 2014] in Fig 9. Also, ROC curves of all other features are given in Fig 9 as plotted using MATLAB's inbuilt perfcurve function which calculates TPR and FPR at various threshold values based on

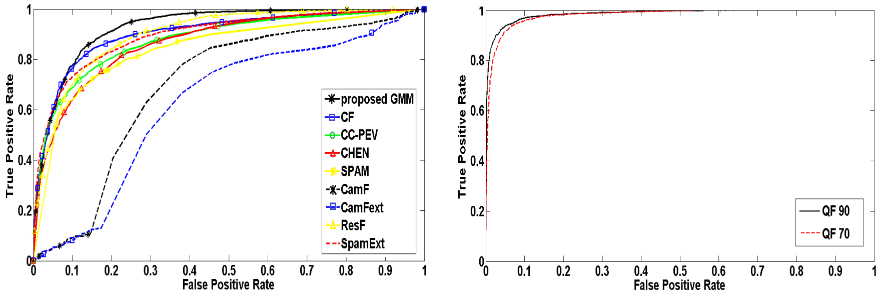

Fig. 9. *Left:* ROC curve using the proposed method, Chen features [Chen and Shi 2008], CC-Pev features [Pevny and Fridrich 2007], CF features [Kodovsky et al. 2012], SPAM features [Kirchner and Fridrich 2010], Residual features [Cozzolino et al. 2014] and Camera based features [Verdoliva et al. 2014] on general filtering detection for quality factors 30. *Right:* ROC curve using the proposed method to detect filtering when two random filtering operations are performed on the image.

the scores produced by the respective classifier for quality factor 30. For example, the TPR of classification for quality factor 30 for an FPR of as low as 0.1 for the proposed method is 0.86. The TPR using CF features is 0.82, CC-Pev features is 0.79, Chen features is 0.77 while SPAM achieves 0.78. ResF has a TPR of 0.8 and SpamExt gives 0.8 for the same FPR. Also, CamF and CamFext achieves a TPR of 0.75 when FPR is around 0.35. More importantly, comparison with SPAMext and CamFext whose dimensionality is greater than the proposed method, shows that the proposed technique's good performance is not entirely because of the high dimensionality. We attribute it to the effectiveness of compression noise in characterizing the modifications the image has undergone. Also, CamF features make an important assumption that the camera with which the image under consideration was taken, is available with us as it is camera based features. It is evident from the above comparisons that the proposed technique is better in detecting filtering under less constraints compared to state of the art filtering detection algorithms and popular forensic features. Also, even when the dimension of similar features like SPAM is extended to what the proposed technique has, our algorithm performs better. It can be noted that the performance of the proposed technique is better in Table VI compared to Sec 5.1. This might be because the dataset considered in Sec 5.1 is a composite of three totally different databases resulting in a larger variance. Also the cardinality of that dataset is relatively lesser than what is considered for Table VI.

6.3. Detecting multiple operations

In order to check if the proposed technique can detect filtering when the image is filtered more than once, we perform a separate experiment. We construct a filtered images dataset from BOSS images like given in section 6.2, only half the images are linearly filtered and the rest are non linearly filtered after their first filtering operation. We have now 5000 images that are filtered twice. We test these images with the model from section 6.2 already trained for filtering detection. The results of the same is given in Fig 9 as a ROC curve for multiple operation detection for QF 70 and QF 90. TPR for both is above 0.9 for an FPR of 0.05 which shows that the proposed technique is efficient in detecting filtering even when more than one filtering operation has been done on the image.

6.4. Localization of Filtering

Detecting image manipulation in low resolution images and localized image windows is necessary when only a part of the image being tested is forged or enhanced. For example, a low pass filtered image can be cropped and pasted over an unfiltered image resulting in a forged image that is enhanced only partly. We test the performance of our algorithm in detecting filtering in low resolution images. The pipeline followed is similar to [Conotter et al. 2013a] as given in section 6.1.1. In addition to 256×256, we train and test on 128×128 and 64×64 patches cropped from the center of the original images separately. The ROC curves for these experiments is given in Fig 10.

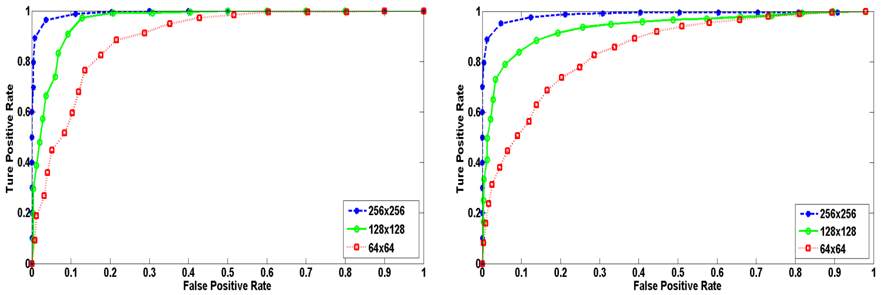

Fig. 10. *Left*: ROC curves of linear filtering detection for 256×256, 128×128 and 64×64 patches using GMM image prior, *Right*: ROC curves of median filtering detection for 256×256, 128×128 and 64×64 patches using GMM image prior.

For 256×256 patches TPR is above 0.97 for FPR of 0.05. It can also be observed that the performance of the algorithm reduces with the reduction in size but we are still able to achieve a TPR of above 0.7 for FPR of 0.05 with 128×128 patches. TPR reaches above 0.9 for a low FPR of just 0.1. This is significant in the case of localization where overlapping 128×128 patches of an image can be tested to localize the filtered part. However performance on 64×64 patches can be seen to be significantly low. Similar experiment for median filtering detection [Kang et al. 2013] is performed as given in section 6.1.2. The ROC curve of this experiment for various sizes is given in Fig 10. Similar observation as that of general filtering detection is made except the performance for 64×64 patches is significantly reduced. The localization experiments were all performed using GMM image prior. It is shown that the proposed algorithm without any modification can be used for detecting filtering, classifying the type of filter applied and localizing the filtered part to an extent.

7. COUNTERING ANTI-FORENSICS

With the advent of anti-forensic techniques such as [Kirchner and Bohme 2008], [Fan et al. 2013] and [Fan et al. 2015], that hides any manipulation performed on an image, it is inevitable to prove the robustness of new forensic algorithms against such techniques. Since the proposed method does not directly depend on the DCT coefficients or the image pixel intensity, it is robust against state-of-the-art anti-forensic algorithms. These anti-forensic techniques drastically reduce the performance of forensic algorithms like [Lai and Bhme 2011] and [Valenzise et al. 2011]. Therefore, it is necessary to evaluate the performance of the proposed algorithm against these techniques.

7.1. JPEG anti-forensics

We implement a popular anti-forensic algorithm proposed by Wei Fan et. al. [Fan et al. 2013]. This algorithm implements intelligent image restoration technique based on Gaussian Mixture prior model to denoise a compressed JPEG image. The removal of compression artifacts and other subtle forgery artifacts make it difficult for general double compression detection algorithms to detect manipulation [Fan et al. 2013]. We perform an experiment wherein 1000 random images from the UCID database are forged and passed through anti-forensic pipeline involving denoising, applying QCS

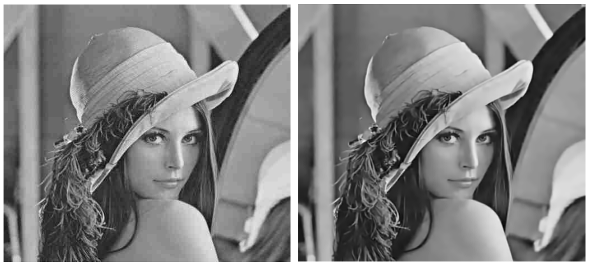

Fig. 11. *Left*: Lena JPEG image of quality factor 20 (PSNR-32.9625 dB) and *Right*: GMM restored version (PSNR-33.2196 dB)

Table VII. Accuracy in detecting anti-forensically forged images as manipulated, using HMRF prior for noise extraction.

QUALITY	ACCURACY (%)	
FACTOR	WITHOUT I.F.	WITH I.F.
30	99.8	100
40	99.0	100
50	99.3	100
60	99.5	100
70	99.6	100
80	99.8	100
90	99.2	100

(Quantization Constrained Set) constraint and calibrating [Fan et al. 2013]. In Fig 11, an example of a compressed image and its restored version using the GMM denoising method in [Fan et al. 2013] is given. It can be seen that the GMM enhanced version has very low blocking artefacts compared to the compressed version.There is a separate Improved Forensics (I.F) step introduced in [Fan et al. 2013] that minimizes a cost function to make the image, undetectable by detectors like [Lai and Bhme 2011] and [Valenzise et al. 2011]. We implement this Improved Forensics step over the normal anti-forensic operations for 100 images separately. HMRF prior based noise and TPM features are extracted from the 1000 anti-forensically operated images without I.F and the 100 images with I.F step. The TPMs are tested against the original model already trained with 3000 filtered and unfiltered samples. The accuracy in detecting these anti-forensically forged images as manipulated, is given in Table VII as the percentage of images classified as manipulated, out of 1000 images in the without I.F case and out of 100 in with I.F case. It can be seen that the percentage of accuracy is as high as 99% and above for images of size 256×256. As the anti-forensic algorithm tries to remove the noise, it still implants subtle artifacts. Further, it may not eliminate compression noise completely and the spatial correlation of the left over noise would not be preserved. This proves the efficacy of the proposed algorithm against intelligent enhancement techniques.

Also, we perform the same detection experiment with low resolution anti-forensically forged images (128×128 and 64×64) similar to the experiments given in section 6.4. The ROC curves of these experiments are given in Fig 12. It can clearly be seen that for an FPR of 0.05, TPR as high as 0.85 is achieved even for 64×64 patches. Next, we implement some of the features that were used in section 6.2 on this anti-forensically modified images and try to detect filtering. The results of this

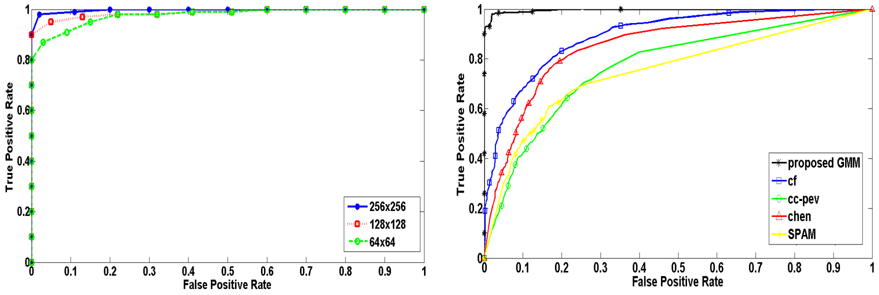

Fig. 12. *Left*: ROC curves for detection of anti-forensically forged images of sizes 256×256, 128×128 and 64×64, ***Right***: ROC curves for detection of JPEG anti-forensically forged images of sizes 256×256 by features such as Chen [Chen and Shi 2008], CC-PEV [Pevny and Fridrich 2007], CF [Kodovsky et al. 2012] and SPAM [Kirchner and Fridrich 2010] for quality factor 90.

experiment is provided as the ROC curve of detection using features [Kirchner and Fridrich 2010], [Chen and Shi 2008], [Pevny and Fridrich 2007] and [Kodovsky et al. 2012]. It is evident from the ROC curve shown in Fig 12 for quality factor $qf_2 = 90$ that all these methods fail to detect filtering in the presence of compression anti-forensics. Their TPR drops to less than 0.6 for an FPR of 0.05 while the proposed method from Fig 12 for 256×256 size still gives a TPR above 0.9. Even though the steganalytic features did not give very low performance in general filtering detection, it can be seen from this experiment that they are all vulnerable to jpeg anti-forensic attack.

7.2. Median filtering antiforensics

It has been shown in the literature [Kirchner and Bohme 2008], [Kirchner and Fridrich 2010] and [Wu et al. 2013] that median filtering is by itself an effective anti-forensic operation. It degrades the performance of many forensic algorithms and thus a lot of median filtering detection techniques were introduced. The non linearity of the median filter makes it difficult for forensic algorithms that utilize linearity assumptions to detect forgery. We implemented a median filtering anti-forensic algorithm [Fan et al. 2015] by Wei Fan et al. that has been able to degrade the performance of most of the median filtering detection algorithms. We considered 2000 images from the 'original' dataset and median filtered 1000 of those images with 3×3 kernel filter. We then implemented the anti-forensic algorithm described in [Fan et al. 2015] over the median filtered images. We then tried classifying the 2000 images as unfiltered or median filtered using our original trained general filtering detection SVM model from section

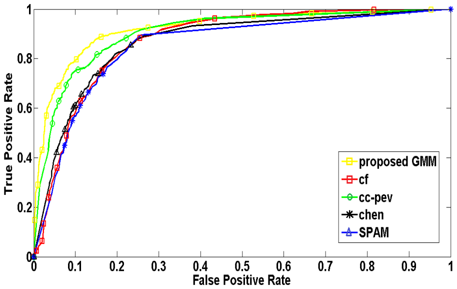

Fig. 13. ROC curve for detection of anti-forensically forged median filtered images of quality factor 90 using proposed method and by features such as Chen [Chen and Shi 2008], CC-PEV [Pevny and Fridrich 2007], CF [Kodovsky et al. 2012] and SPAM [Kirchner and Fridrich 2010].

5.1. The accuracy of classification for quality factor qf_2 90 is 86.5% whereas that of quality factor 30 is 72%. The ROC curve of detection for quality factor 90 is given in Fig 13. This proves the efficacy of the proposed technique against median filtering anti-forensics. However, when we use the model trained for 'median filtering detection' from section 6.1.2, the anti-forensic algorithm succeeds in bringing down the accuracy in detection to less than 60%.

We also compare the results for this detection with some of the features from section 6.2 for detecting median filtering under median filtering antiforensics using 'general filtering detection' model as explained in the same section. The results are given as ROC curve for quality factor 90 in Fig 13. It can be noted that the proposed technique performs better than any of the popular features. For lower quality factors, the performance of these features reduced drastically whereas that of the proposed method remained at above 70%.

8. DISCUSSION

In the proposed technique two natural image models are used owing to their good performance in image restoration ([Fan et al. 2013], [Li and Singh 2009]). However, we imply that any other image prior model can be used and better models may yield better results. In order to give a comparison as to which of the two image priors used in the paper is better, factors such as speed, efficiency and flexibility are to be taken into account. In the HMRF image prior based method, the amount of noise extracted can be controlled by parameters such as weight, learning rate, smoothing parameter etc. thereby providing scalability. However, the time taken by HMRF prior based denoising followed by TPM extraction is approximately 120 seconds per image on a computer with Intel i3 processor and 4GB RAM. This is six times that of what GMM prior based denoising followed by TPM extraction takes. This is because GMM based denoising involves a one step MAP approximation rather than an extensive gradient descent. Also the dimensionality of the TPM features is reduced in the GMM prior based experiments because of lower variance in the compression noise extracted. Still, GMM prior based denoising does not provide scalability to modify the denoising parameters since it is learning based. We believe either of the priors must mostly give similar results even though GMM based is faster.

9. CONCLUSION

This paper presents a novel quantization noise based method to detect and classify filtering operations applied on an image as part of forgery. Towards this, we derive a modification of the HMRF image model which can take blocking artifacts into account. We then investigated transition probability features of quantization noise for detecting and classifying the filters. Further, we also study the effectiveness of GMM prior model and compare the results for both the prior distributions. The robustness of the proposed scheme is demonstrated through analytical and empirical measures.

The following inferences are made from the results obtained using the proposed method. *First*, the detection and classification accuracies are above 80% and around 88% respectively. This shows that the transition probabilities of the compression noise of the image or its modified version are robust measure of classifying image manipulations. *Second*, the method gives high accuracies for detection and classification under a wide range of Quality Factors and filters. Also, the proposed technique is effective in detecting filtering enhancement in low resolution images and images that are filtered more than once. In addition, the algorithm can potentially be used for localizing the enhanced regions. *Third*, this technique is effective against anti-forensically manipulated images giving a very high accuracy against both compression based anti-forensics and median filter anti-forensics. *Finally*, to the best of our knowledge, we believe that, this is the first approach which targets filtering detection in both JPEG and TIFF images under various real world experimental settings and is effective against popular anti-forensic techniques.

Future work will be to analyze the effect of contrast enhancement along with filtering enhancement on compression noise. Also, how effectively can quantization noise be used for localization of forgery will be investigated.

REFERENCES

P. Bas, T. Filler, and T. Pevny. 2011. Break Our Steganographic System: The ins and outs of Organizing BOSS. In *Proc. 13th Int. Conf. on Information Hiding, Lecture Notes in Computer Science*, Vol. 6958. 59–70.

Gaurav Bhatnagar, Q. M. Jonathan Wu, and Pradeep K. Atrey. 2013. Secure Randomized Image Watermarking Based on Singular Value Decomposition. *ACM Trans. Multimedia Comput. Commun. Appl.* 10, 1 (dec 2013), 4:1–4:21.

Hong Cao and A.C. Kot. 2012. Manipulation Detection on Image Patches Using FusionBoost. *IEEE Transactions on Information Forensic and Security* 7 (June 2012), 992–1002. Issue 3.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology* 2 (2011), 27:1–27:27. Issue 3. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

Chenglong Chen, Jiangqun Ni, and Jiwu Huang. 2013. Blind Detection of Median Filtering in Digital Images: A Difference Domain Based Approach. *IEEE Transactions on Information Forensic and Security* 22 (December 2013), 4699–4710. Issue 12.

C. Chen and Y. Q. Shi. 2008. JPEG Image Steganalysis Utilizing both Intrablock and Interblock Correlations. In *Proc. IEEE Int. Symp. on Circuits and Systems*. 3029–3032.

Chunhua Chen, Yun Q. Shi, and Wei. Su. 2008. A Machine Learning Based Scheme for Double JPEG Compression Detection. In *Proc. IEEE Int. Conf. on Pattern Recognition*. 1–4.

V. Conotter, P. Comesana, and F. Perez-Gonzalez. 2013a. Forensic analysis of full-frame linearly filtered JPEG images. In *Proc. IEEE Int. Conf. on Image Processing*. 4517–4521.

V. Conotter, P. Comesana, and F. Perez-Gonzalez. 2013b. Joint detection of full-frame linear filtering and JPEG compression in digital images. In *Proc. IEEE Int. Workshop on Information Forensics ans Security*. 156–161.

D. Cozzolino, D. Gragnaniello, and L. Verdoliva. 2014. Image Forgery detection through residual-based local descriptors and block-matching. In *Proc. IEEE Int. Conf. on Image Processing*. 5297–5301.

Wei Fan, Kai Wang, François Cayre, and Zhang Xiong. 2013. JPEG Anti-forensics Using Non-parametric DCT Quantization Noise Estimation and Natural Image Statistics. In *Proc. of the First ACM Workshop on Information Hiding and Multimedia Security*. ACM, 117–122.

W. Fan, K. Wang, F. Cayre, and Z. Xiong. 2015. Median Filtered Image Quality Enhancement and Anti-Forensics via Variational Deconvolution. *IEEE Transactions on Information Forensic and Security* 10 (April 2015), 1076–1091. Issue 5.

H. Farid. 2009. Exposing Digital Forgeries From JPEG Ghosts. *IEEE Transaction on Information Forensics and Security* 4 (March 2009), 154–160. Issue 1.

Jessica Fridrich and Jan Kodovsky. 2012. Rich models for steganalysis of digital images. *IEEE Transactions on Information Forensic and Security* 7, 3 (2012), 868–882.

T. Gloe and R. Bohme. 2010. The Dresden Image Database for Benchmarking Digital Image Forensics. *Journal of Digital Forensic Practice* 3 (2010), 150–159. Issue 2-4. available at http://forensics.inf.tu-dresden.de/ddimgdb/.

Kaiming. He, Jian Sun, and Xiaoou Tang. 2013. Guided Image Filtering. *IEEE Tran. on Pattern Analysis and Machine Intelligence* 35, 6 (June 2013), 167–200.

Fangjuan. Huang, Jiwu. Huang, and Y.Q. Shi. 2010. Detecting Double JPEG Compression With the Same Quantization Matrix. *IEEE Transaction on Information Forensics and Security* 5 (2010), 848–856. Issue 4.

Xiangui Kang, M.C. Stamm, Anjie Peng, and K.J.R. Liu. 2013. Robust Median Filtering Forensics Using an Autoregressive Model. *IEEE Transactions on Information Forensic and Security* 8 (September 2013), 1456–1468. Issue 9.

M. Kirchner and R. Bohme. 2008. Hiding Traces of Resampling in Digital Images. *IEEE Transactions on Information Forensic and Security* 3 (November 2008), 582–592. Issue 4.

M. Kirchner and J. Fridrich. 2010. On detection of median ltering in digital images. In *Proc. SPIE, Electron. Imaging, Media Forensics and Security II*, Vol. 7541. 112.

J. Kodovsky, J. Fridrich, and V. Holub. 2012. Ensemble Classifiers for Steganalysis of Digital Media. *IEEE Transactions on Information Forensic and Security* 7, 2 (April 2012), 432–444.

S. Lai and R. Bhme. 2011. Countering Counter Forensics: the case of JPEG compression. In *Proc. Int. Conf. on Information Hiding*. 285–298.

Ankita Lathey and Pradeep K. Atrey. 2015. Image Enhancement in Encrypted Domain over Cloud. *ACM Trans. Multimedia Comput. Commun. Appl.* 11, 3 (feb 2015), 38:1–38:24.

S.Z. Li and S. Singh. 2009. Markov Random field modeling in Image Analysis. *Springer* 3 (2009).

S. Z. Li. 1995. *Markov Random Field Modeling in Computer Vision*. Springer-Verlag, London, UK, UK.

Qingzhong. Liu, A.H. Sung, and Mengyu. Qiao. 2011. A Method to Detect JPEG-Based Double Compression. *Advances in Neural Networks ISNN Springer 2011* 6676 (2011), 466–476.

Q. Liu, A. H. Sung, M. Qiao, Z. Chen, and B. Ribeiro. 2010. An Improved Approach to Steganalysis of JPEG Images. *Informaton Sciences* 180 (2010), 1643–1655. Issue 9. Software available at http://www.shsu.edu/~qxl005/New/Downloads.

T. Pevny, P. Bas, and J. Fridrich. 2010. Steganalysis by Subtractive Pixel Adjacency Matrix. *IEEE Transactions on Information Forensic and Security* 5 (June 2010), 215–224. Issue 2.

T.. Pevny and J. Fridrich. 2007. Merging Markov and DT features for Multiclass JPEG Steganalysis. In *Proc. SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents*. 3 1–3 14.

G. Puglisi, A.R. Bruna, F. Galvan, and S. Battiato. 2013. First JPEG quantization matrix estimation based on histogram analysis. In *Proc. IEEE Int. Conf. on Image Processing*. 4502–4506.

Xiaoqing Qiu, Haodong Li, Weiqi Luo, and Jiwu Huang. 2014. A Universal Image Forensic Strategy Based on Steganalytic Model. In *Proceedings of the 2nd ACM Workshop on Information Hiding and Multimedia Security (IHMMSec '14)*. ACM, 165–170.

H. Ravi, A.V. Subramanyam, B. Avinash kumar, and G. Gupta. 2014. Compression Noise Based Video Forgery Detection. In *Proc. IEEE Int. Conf. on Imagel Processing*. 5352–5356.

H. Ravi, A. V. Subramanyam, and S. Emmanuel. 2015. Spatial Domain Quantization Noise Based Image filtering Detection. In *Proc. IEEE Int. Conf. on Image Processing*.

M.A Robertson and R.L. Stevenson. 2005. DCT quantization in Compressed Images. *IEEE Transactions on Circuits and Systems for Video Technology* 13 (January 2005), 27–38.

G. Schaefer and M. Stich. 2004. UCID - An Uncompressed Colour Image Database. In *Proc.SPIE Storage and Retrieval Methods and Applications for Multimedia*. 472–480.

M. C. Stamm, M. Wu, and K. J . R. Liu. 2013. Information Forensics: An Overview of the First Decade. *IEEE Access* 1 (May 2013), 167–200.

G. Valenzise, V. Nobile, M. Tagliasacchi, and S. Tubaro. 2011. Countering JPEG anti-forensics. In *Proc. IEEE Int. Conf. on Image Processing*. 1949–1952.

L. Verdoliva, D. Cozzolino, and G. Poggi. 2014. A feature-based approach for image tampering detection and localization. In *Proc. IEEE Int. Workshop on Information Forensics and Security*. 149–154.

Robert Ware and Frank Lad. 2003. Approximating the distribution for sums of products of normal variables. *University of Canterbury, England, Tech. Rep. UCDMS* 15 (2003), 2003.

Zhung-Han Wu, M.C. Stamm, and K.J.R. Liu. 2013. Anti-forensics of Median Filtering. In *Proc. IEEE Int. Conf. on Acoustics Speech and Signal Processing*. 3043–3047.

Hui Zeng, Tengfei Qin, Xiangui Kang, and Li Liu. 2014. Countering anti-forensics of median filtering. In *Proc. IEEE Int. Conf. on Acoustics Speech and Signal Processing*. 2704–2708.

Y. Zhang, S. Li, S. Wang, and Y.Q. Shi. 2014. Revealing the Traces of Median Filtering Using High-Order Local Ternary Patternsl. *IEEE Signal Processing Letters* 21 (March 2014), 275–280. Issue 3.

D. Zoran and Y. Weiss. 2011. From learning models of natural image patches to whole image restoration. In *Proc. IEEE Int. Conf. on Computer Vision*. 479–486.