

Normalized Energy Density-Based Forensic Detection of Resampled Images

Xiaoying Feng, *Member, IEEE*, Ingemar J. Cox, *Fellow, IEEE*, and Gwenaél Doërr, *Member, IEEE*

Abstract—We propose a new method to detect resampled imagery. The method is based on examining the normalized energy density present within windows of varying size in the second derivative of the image in the frequency domain, and exploiting this characteristic to derive a 19-D feature vector that is used to train a SVM classifier. Experimental results are reported on 7500 raw images from the BOSS database. Comparison with prior work reveals that the proposed algorithm performs similarly for resampling rates greater than 1, and is superior to prior work for resampling rates less than 1. Experiments are performed for both bilinear and bicubic interpolations, and qualitatively similar results are observed for each. Results are also provided for the detection of resampled imagery with noise corruption and JPEG compression. As expected, some degradation in performance is observed as the noise increases or the JPEG quality factor declines.

Index Terms—Image forensics, normalized energy density, resampling detection.

I. INTRODUCTION

FORENSIC signal processing attempts to identify the variety of processing steps that a signal has undergone. Such information is useful in determining whether, for example, a signal is authentic or has been tampered with. There are two main approaches to multimedia forensics: active forensics and passive forensics [1]. Active forensics relies on modifying the multimedia signal prior to its distribution to assist in later forensic analysis. Digital watermarks [2] are one example of active forensics. A limitation of active forensics is the need for content-generating devices, e.g. cameras, sensors, microphones, that embed watermarks. Such devices are often not available, and in these cases active forensics cannot be applied. Passive forensics, in contrast, does not rely on any prior modification of the signal. As such, passive forensics is, in theory, applicable to a broader range of operating scenarios.

A variety of passive forensic methods have been developed to detect, for example, requantization [3]–[7], resampling [8]–[14], and affine transformation [15], [16]. In this paper, we

consider the problem of determining whether an image¹ has undergone resampling. Section II provides a review of prior work on this topic. Section III describes our normalized energy density-based method. Section IV reports experimental results performed using the BOSS version 0.9 database [17] consisting of 7500 raw images. Comparison is made with the algorithm described in [9]. Finally, Section V summarizes our results and discusses possible directions for future work.

II. PRIOR WORK

Early work on detecting image resampling was based on the observation that artifacts were introduced in the resampled imagery due to interpolation, a basic operation involved in resampling. Generally, these artifacts are periodic in the spatial domain and therefore manifest themselves as peaks in the corresponding frequency domain.

Popescu and Farid [8] noted that the interpolation process introduces correlations between the resampled imagery pixels. They proposed measuring these correlations based on an expectation/maximization (EM) algorithm. The EM algorithm estimates the “average” correlation that is present in the image and subsequently computes the probability of the pixels being correlated to their neighbors. The corresponding correlation probability map (p-map) in the DFT domain exhibits periodic peaks that are not present in single-sampled images.

The work of Popescu and Farid was subsequently refined. Mahdian and Saic [9] proposed an automatic detector that searches for local maxima, which are defined as n times greater than a local average magnitude. Kirchner [10], [11] replaced the EM algorithm, which is computationally demanding, with linear filtering [10] and linear row and column predictors [11], and proposed an automatic detector based on the maximum gradient of the p-map’s spectrum. Dalgaard *et al.* [12], and Vazquez-Padín and Pérez-González [13] exploited the underlying cyclostationarity of communication signals. Based on cyclostationarity theory, they examined a series of prefilters in order to improve the detection accuracy of imagery resampling. More recently, Feng *et al.* [14] proposed a new normalized energy density-based method for resampling detection. In this paper, we further investigate this method from both the theoretical and experimental perspectives.

The method described in [9] is used as the baseline against which we compare our algorithm. The second derivative is taken along either the horizontal or the vertical dimension of an image. The radon transformation then projects the second derivative to each of 180 directions, where the projection angles are integers

Manuscript received October 02, 2011; revised October 02, 2011; accepted March 01, 2012. Date of publication April 03, 2012; date of current version May 11, 2012. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Dinei A. Florencio.

X. Feng and I. J. Cox are with the Department of Computer Science, University College London, London WC1E 6BT, U.K. (e-mail: x.feng@cs.ucl.ac.uk).

G. Doërr is with the Technicolor R&D, Security and Content Protection Labs, 35576 Cesson-Sévigné Cedex, France.

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

Digital Object Identifier 10.1109/TMM.2012.2191946

¹For the sake of simplicity, we only discuss grey-scale images in this paper. However, a color image can be represented by three channels, i.e., intensity, saturation and hue. Our proposed method can be directly applied in the intensity channel of color images.

from 0° to 179° . The detection of resampling is based on the detection of periodicity in the autocovariance of the projected vectors. In so doing, the first derivative of all 180 projected vectors are calculated. The autocovariance of the first derivative is then computed. Finally, the periodicity of autocovariance is detected in the DFT domain, using a local maxima detector. Further details of the algorithm can be found in [9].

III. NORMALIZED ENERGY DENSITY-BASED METHOD

Our method for detecting resampling is based on a 19-D feature vector that is derived from an examination of the normalized energy density present at various window sizes in the DFT of the second-derivative of an image.

A. Normalized Energy Density

Let $\mathbf{i}(x, y)$ denote an image, and $\mathbf{I}(u, v)$ denote its frequency spectrum, i.e., the DFT of the image. We assume, for simplicity and without loss of generality, that the image is a square with dimension n , i.e., there are n^2 pixels present in the image.

Using Parseval's equation, the energy, E , present in an image, is given by

$$E = \sum_{x=0}^{n-1} \sum_{y=0}^{n-1} \mathbf{i}(x, y)^2 = \sum_{u=-w_c}^{w_c} \sum_{v=-w_c}^{w_c} |\mathbf{I}(u, v)|^2 \quad (1)$$

where w_c is the cutoff frequency of the image. We define $E(w)$ to be the energy present in the power spectrum, in a window of dimension $\pm w$, where $0 \leq w \leq w_c$

$$E(w) = \sum_{u=-w}^w \sum_{v=-w}^w |\mathbf{I}(u, v)|^2. \quad (2)$$

The energy density, $E_d(w)$, is then defined by the averaged energy within the window ($2w \times 2w$), i.e.,

$$\begin{aligned} E_d(w) &= \frac{1}{4w^2} E(w) \\ &= \frac{1}{4w^2} \sum_{u=-w}^w \sum_{v=-w}^w |\mathbf{I}(u, v)|^2. \end{aligned} \quad (3)$$

To accommodate for the likely difference of dimensions between the original image and its resized versions, we normalize the window size, s , with respect to the cutoff frequency, w_c , i.e.,

$$s = \frac{w}{w_c} \quad (4)$$

where s takes values between 0 and 1. As a result, the corresponding **normalized energy density**, $E_n(s)$, which is independent of the size of the image under investigation, is given by

$$\begin{aligned} E_n(s) &= \frac{E_d(w_c \cdot s)}{E_d(w_c)} \\ &= \frac{1}{s^2} \frac{\sum_{u=-s \cdot w_c}^{s \cdot w_c} \sum_{v=-s \cdot w_c}^{s \cdot w_c} |\mathbf{I}(u, v)|^2}{\sum_{u=-w_c}^{w_c} \sum_{v=-w_c}^{w_c} |\mathbf{I}(u, v)|^2} \end{aligned} \quad (5)$$

where $E_d(w_c)$ represents the energy density of the whole image.

B. Image Resampling

Let $\mathbf{i}_o(x, y)$ and $\mathbf{i}_r(x, y)$ denote the original image and its corresponding resampled version respectively. Mathematically, resampling can be represented as the convolution of the single-sampled image, $\mathbf{i}_o(x, y)$, and the impulse response function (IRF) of the resampling system, $\mathbf{h}_r(x, y)$, that corresponds to the specific interpolation method and any post-processing being used, i.e.,

$$\mathbf{i}_r(x, y) = \mathbf{i}_o(x, y) * \mathbf{h}_r(x, y). \quad (6)$$

Correspondingly, in the frequency domain, we have

$$\mathbf{I}_r(u, v) = \mathbf{I}_o(u, v) \cdot \mathbf{H}_r(u, v). \quad (7)$$

In practice, bilinear interpolation and bicubic interpolation are two widely-used interpolation methods. Specifically, bilinear interpolation considers the closest 2×2 neighborhood of known pixel values surrounding the interpolated (unknown) pixel. It then takes a weighted average of these four pixels to calculate the interpolated value. In contrast, bicubic interpolation considers the closest 4×4 neighborhood of known pixels. Similarly, it then takes a weighted average of these 16 pixels to calculate the interpolated value. In this paper, both bilinear and bicubic interpolations are considered for comparative purposes.

Note that anti-aliasing filters are routinely applied as a post-processing after down-sampling an image to avoid undesired visual artifacts. Therefore, anti-aliasing option is activated with the MATLAB function `imresize()` to perform the resampling operations in the experiments in Section IV. However, for the theoretical analysis of resampling, we do not consider any post-processing in this Section.

C. Up-Sampling and Down-Sampling

Suppose that the cutoff frequency of a single-sampled image is w_c . Note that an original image and its resized versions will not have the same dimensions. In the frequency domain, these different dimensions will be represented by different cutoff frequencies. For up-sampled images, i.e., resampling rate $\rho > 1$, the resampled images will have larger dimensions than the single-sampled images. In other words, the cutoff frequencies of up-sampled images are larger than that of the single-sampled images. Similarly, the cutoff frequencies of down-sampled images, i.e., $\rho < 1$, are smaller than those of corresponding single-sampled images.

Mathematically, the cutoff frequency of a resampled image with resampling rate ρ should be $\rho \cdot w_c$. In other words, the frequency spectrum of a resampled image is between $-\rho \cdot w_c$ and $\rho \cdot w_c$. Therefore, in order to generate a resampled image, it is equivalent for the single-sampled image to go through a low-pass filter with the cutoff frequency $\rho \cdot w_c$. For the purpose of theoretical analysis, we consider an ideal low-pass filter, i.e.,

$$\mathbf{H}_r(u, v) = \begin{cases} 1, & \text{if } 0 \leq u, v \leq \rho \cdot w_c \\ 0, & \text{otherwise} \end{cases}. \quad (8)$$

1) *Up-Sampling*: Based on the ideal low-pass filter shown in (8), the width of the frequency spectrum of an up-sampled

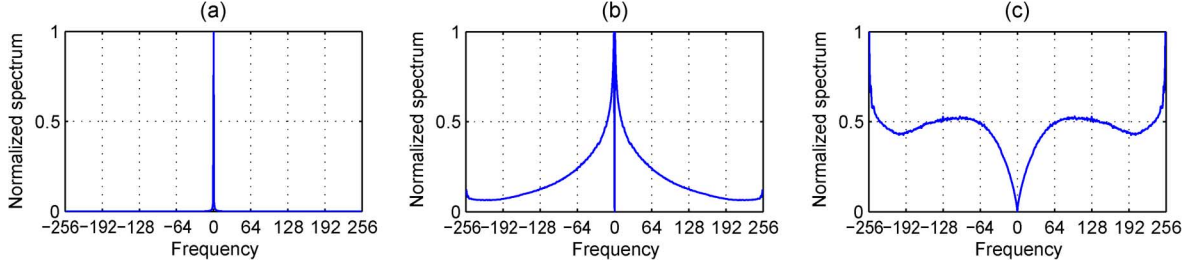


Fig. 1. Normalized spectrum in the diagonal direction (i.e., $u = v$). (a) Without high-pass filter. (b) After first-derivative high-pass filter. (c) After second-derivative high-pass filter. The curves are averaged over 7500 images from the BOSS database.

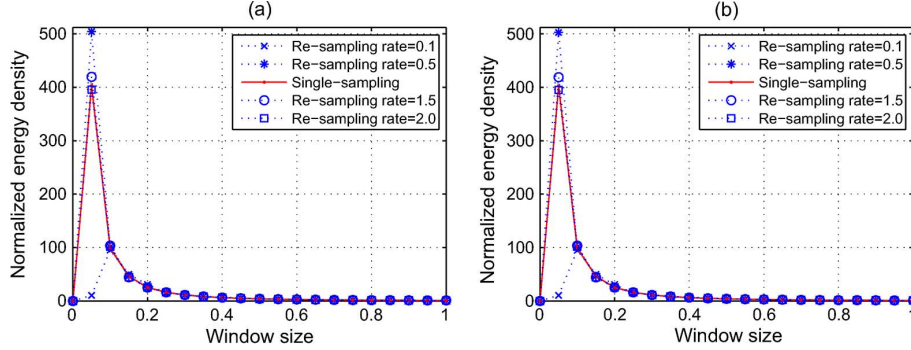


Fig. 2. Normalized energy density curves for a set of resampling rates, averaged over 7500 images from the BOSS database (without high-pass filtering, without anti-aliasing, fixed content cropping). (a) Bilinear interpolation. (b) Bicubic interpolation.

image, $\mathbf{I}_r(u, v)$, is extended from $0 \leq u, v \leq w_c$ to $0 \leq u, v \leq \rho.w_c$ ($\rho > 1$), i.e.,

$$\mathbf{I}_r(u, v) = \begin{cases} \mathbf{I}_o(u, v), & \text{if } 0 \leq u, v \leq w_c \\ 0, & \text{if } w_c < u \text{ or } v \leq \rho.w_c \end{cases} \quad (9)$$

Therefore, the energy density of an up-sampled image, $E_{dr}(w)$, is given by

$$E_{dr}(w) = \begin{cases} E_{do}(w), & \text{if } 0 \leq w \leq w_c \\ \frac{w_c^2}{w^2} E_{do}(w_c), & \text{if } w_c < w \leq \rho.w_c \end{cases} \quad (10)$$

where $E_{do}(w)$ and $E_{dr}(w)$ represent the energy density of the single-sampled and resampled images respectively.

2) *Down-Sampling*: Based on the ideal low-pass filter shown in (8), the width of the frequency spectrum of a down-sampled image, $\mathbf{I}_r(u, v)$, is narrowed from $0 \leq u, v \leq w_c$ to $0 \leq u, v \leq \rho.w_c$ ($\rho < 1$), i.e.,

$$\mathbf{I}_r(u, v) = \mathbf{I}_o(u, v) \quad 0 \leq u, v \leq \rho.w_c. \quad (11)$$

Therefore, the energy density of a down-sampled image, $E_{dr}(w)$, is given by

$$E_{dr}(w) = E_{do}(w), \quad 0 \leq w \leq \rho.w_c. \quad (12)$$

D. High-Pass Filtering

It is well-known that the spectrum present in an image is typically concentrated in the lower frequencies, and drops off with increasing frequency. The shape of the spectrum is often approximated by a Gaussian model. Fig. 1(a) shows the normalized spectrum in the diagonal direction, i.e., $|\mathbf{I}(u, u)|$, averaged over all 7500 images in the BOSS database.

Note that the high concentration of energy in the lower frequencies can mask high frequency effects. Fig. 2 shows the normalized energy density curves for a set of different resampling rates, over 7500 images from the BOSS database for window sizes s ranging from 0 to 1, including: 1) single-sampled images (solid curve); 2) resampled images up-sampled at rates 1.5 and 2.0; and 3) resampled images down-sampled at rates 0.1 and 0.5. It is clear that the energy curves of different resampling rates cannot be distinguished from each other, due to their similar strong energy in the lower frequencies. We also observe a “discrepancy”, where the normalized energy density for down-sampled imagery with resampling rate 0.1 decreases. This “discrepancy” appears to be caused by the high concentration of energy in the lower frequencies, and is absent when high-pass filtering is performed.

In order to eliminate the effect of the high energy concentration in the lower frequencies, an image, $\mathbf{i}(x, y)$, is first high-pass filtered before examining its power spectrum. The first-derivative gradient filter and **second-derivative Laplacian filter** are widely-used high-pass filters. It is interesting to compare the effect of these two high-pass filters. Fig. 1(b) and (c) compare the normalized spectrum in the diagonal direction averaged over all 7500 images in the database we used. We can see that **strong low-frequency components remain in the spectrum after the first-derivative high-pass filter**, although the dc element equals 0. In contrast, low-frequency components are significantly reduced by the second-derivative high-pass filter.

For the normalized energy density curves, we would expect that the energy curves for different resampling rates cannot be easily separated after the first-derivative high-pass filtering, due to the remaining strong low-frequency spectrum. In contrast, it is expected that the energy curves for different resampling rates would be well distinguished after second-derivative high-pass

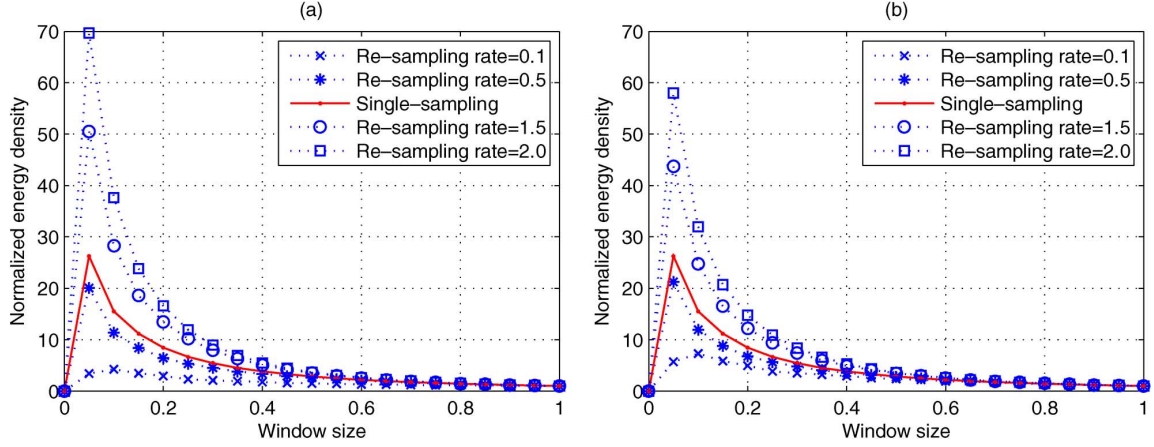


Fig. 3. Normalized energy density curves for a set of resampling rates, averaged over 7500 images from the BOSS database (after first-derivative high-pass filtering, without anti-aliasing, fixed content cropping). (a) Bilinear interpolation. (b) Bicubic interpolation.

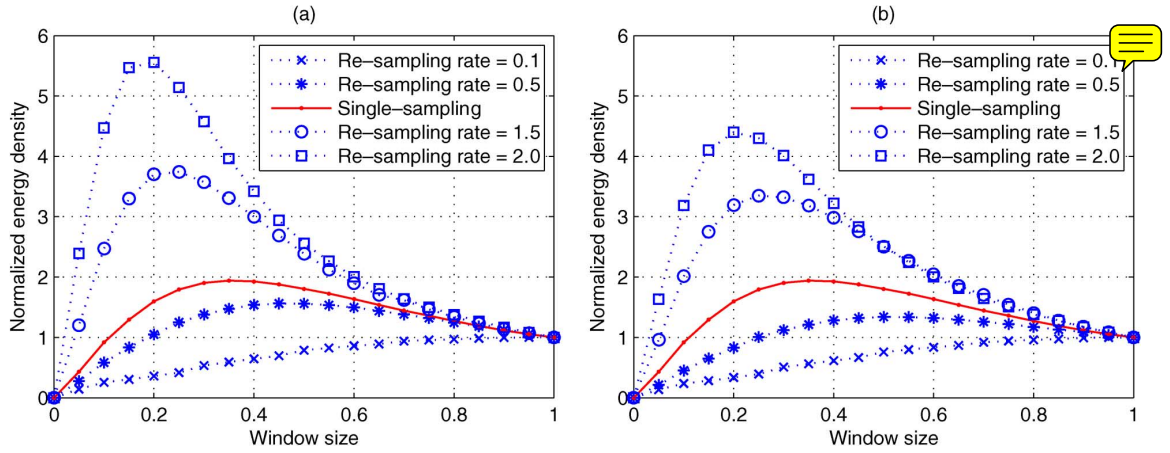


Fig. 4. Normalized energy density curves for a set of resampling rates, averaged over 7500 images from the BOSS database (after second-derivative high-pass filtering, without anti-aliasing, fixed content cropping). (a) Bilinear interpolation. (b) Bicubic interpolation.

filtering since it suppresses more of the low-frequencies. Figs. 3 and 4 show the normalized energy density curves for the images with a set of different resampling rates, after the first-derivative and the second-derivative high-pass filters respectively. The reported energy curves clearly support the discussion above.

We therefore use a second-derivative filter. Specifically, the following Laplacian high-pass filter with the kernel \mathbf{k} , is used in our experiments

$$\mathbf{k} = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}. \quad (13)$$

E. Peak Value and Its Location

The normalized energy density curves of Fig. 4, are visually well separated for different resampling rates. Specifically, for the single-sampled imagery, we see a curve originating at zero (since there is no energy in a window size of zero), increasing monotonically to a peak at a window size of approximately $s = 0.35$, and then monotonically decreasing to a value of 1 when the window size encompasses the entire power spectrum ($s = 1$). If we now compare this to the up-sampled images, we observe a qualitatively similar curve, but the peak values are shifted to the left, i.e., the peak values occur for $s < 0.35$. Similarly for the down-sampled images, we observe a qualitatively similar curve, but the peak values are shifted to the right, i.e.,

the peak values occur for $s > 0.35$. That is, the peak value and its location in the normalized energy density curve are shifted as the resampling rate changes. We theoretically explain this relationship next.

Let us use $E_{nr}(\cdot)$ and $E_{no}(\cdot)$ to represent the normalized energy density of the resampled and single-sampled images respectively. Suppose the peak value in the normalized energy density curve of a single-sampled image occurs at the window size $s = \theta$ ($0 \leq \theta \leq 1$), we have

$$\begin{aligned} E_{no}(s)|_{s=\theta} &= E_{no}(\theta) \\ &= \frac{E_{do}(w_c \cdot \theta)}{E_{do}(w_c)}. \end{aligned} \quad (14)$$

Given (10) and (12), we can replace $E_{do}(w_c \cdot \theta)$ with $E_{dr}(w_c \cdot \theta)$, since the peak in the curve does not occur for a value of w between $w_c < w \leq \rho w_c$ when up-sampling occurs. Thus we have

$$E_{no}(s)|_{s=\theta} = \frac{E_{dr}(w_c \cdot \theta)}{E_{do}(w_c)} \quad (15a)$$

$$= \frac{E_{dr}(\rho \cdot w_c)}{E_{do}(w_c)} \cdot \frac{E_{dr}\left(\rho \cdot w_c \cdot \frac{\theta}{\rho}\right)}{E_{dr}(\rho \cdot w_c)} \quad (15b)$$

$$= \frac{E_{dr}(\rho \cdot w_c)}{E_{do}(w_c)} \cdot E_{nr}\left(\frac{\theta}{\rho}\right) \quad (15c)$$

$$= \frac{E_{dr}(\rho \cdot w_c)}{E_{do}(w_c)} \cdot E_{nr}(s)|_{s=\frac{\theta}{\rho}}. \quad (15d)$$

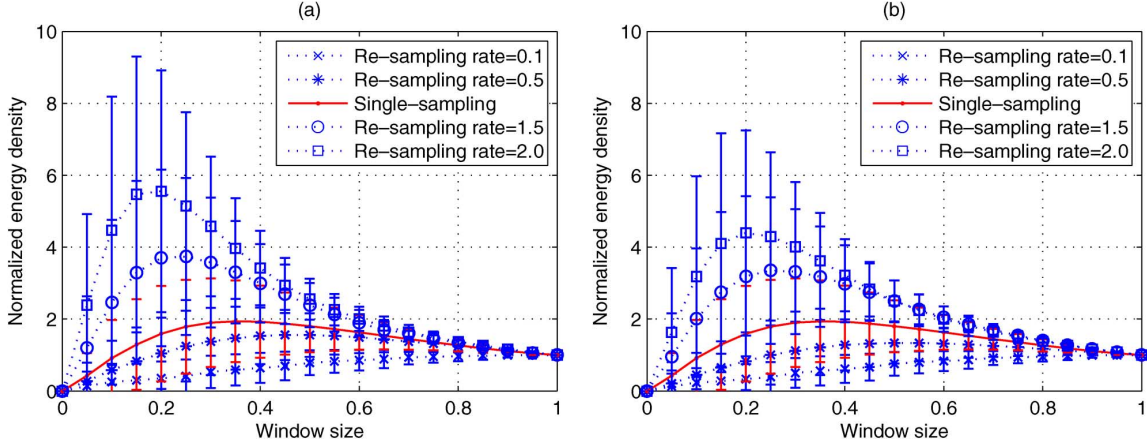


Fig. 5. Normalized energy density curves for a set of resampling rates, averaged over 7500 images from the BOSS database (after second-derivative high-pass filtering, without anti-aliasing, fixed content cropping). The error bars indicate the standard deviations over 7500 images. (a) Bilinear interpolation. (b) Bicubic interpolation.

The derivation from (15a) to (15d) relies on (10) ($0 \leq w_c \leq w_c$) for up-sampling and on (12) for down-sampling. As a result, for a resampled image, the peak value of the normalized energy density curve is shifted to the window size $s = (\theta/\rho)(0 < \theta < 1)$.

1) *Up-Sampling*: For up-sampling, i.e., $\rho > 1$, we have $(\theta/\rho) < \theta$. Therefore, the peak of the normalized energy density curve is shifted to the left. This is to be expected, because for up-sampled imagery, we would expect that less energy is present in the very high frequencies, as these frequencies are absent in the original images. Furthermore, the peak value after up-sampling is given by

$$\begin{aligned} E_{nr} \left(\frac{\theta}{\rho} \right) &= \frac{E_{do}(w_c)}{E_{dr}(\rho \cdot w_c)} \cdot E_{no}(\theta) \\ &= \frac{E_{do}(w_c)}{\frac{w_c^2}{\rho^2 \cdot w_c^2} \cdot E_{do}(w_c)} \cdot E_{no}(\theta) \\ &= \rho^2 \cdot E_{no}(\theta). \end{aligned} \quad (16)$$

It is clear that the peak value is increased after up-sampling.

2) *Down-Sampling*: For down-sampling, i.e., $\rho < 1$, we have $(\theta/\rho) > \theta$. Therefore, the peak of the normalized energy density curve is shifted to the right. This is to be expected, because for down-sampled imagery, the fact that we have more energy in the higher frequencies indicates that these images were derived from images containing higher frequency information. Furthermore, the peak value after down-sampling is given by:

$$\begin{aligned} E_{nr} \left(\frac{\theta}{\rho} \right) &= \frac{E_{do}(w_c)}{E_{dr}(\rho \cdot w_c)} \cdot E_{no}(\theta) \\ &= \frac{E_{do}(w_c)}{E_{do}(\rho \cdot w_c)} \cdot E_{no}(\theta). \end{aligned} \quad (17)$$

Note that (17) is only applicable for $0 \leq (\theta/\rho) \leq 1$ (i.e., $0 \leq \theta \leq \rho \leq 1$), since the possible value of a window size, θ/ρ , should be between 0 and 1. In this case, the fraction in (17) is less than one and the peak value is therefore decreased after down-sampling.

However, if the down-sampling rate, ρ , is smaller than θ , i.e., $0 \leq \rho < \theta \leq 1$, the energy density curve is expected as monotonically increasing to the peak value one at the window size of one.

F. Detection Method

The curves in Fig. 4 suggest that it is possible to differentiate single-sampled and resampled imagery based on the normalized energy density present at various window sizes in the DFT of the second-derivative of the scrutinized images. Fig. 5 shows the normalized energy density curves of Fig. 4 with associated error bars of ± 1 standard deviation, computed over the 7500 images. Clearly, the standard deviations are relatively large, sometimes exhibiting considerable overlap between curves. Consequently, it is not possible to discriminate single-sampling and resampling based on a single peak energy value. In order to overcome this problem, we use a 19-D vector, the values of each dimension being the normalized energy density, $E_n(s)$, for window sizes s ranging from 0.05 to 0.95 in steps of 0.05. It is this vector that is used in the experiments reported next.

IV. EXPERIMENTAL RESULTS

Our experiments use the BOSS database version 0.9 [17], which consists of 7500 raw images, i.e., the images have never undergone resampling. Experiments are performed for both bilinear and bicubic interpolations. Note that both interpolation and anti-aliasing are applied to the images, as this is a more realistic operating scenario.

The 19-D vector was used as input to train support vector machine (SVM) classifiers [18]. The widely-used SVM kernel of radial basis function (RBF) is used. The reported results are the average of 10 trials by applying random subsampling validation, where for each of the 10 trials, training is performed on a random 20% subset of the database, and testing on the remaining 80%.

Instead of manually setting a fixed threshold, a receiver operating characteristics (ROC) curve [19], the threshold of which is varied, is used to evaluate our detection results. Moreover, in order to summarize the classification performance with a single scalar number, we report the area under the ROC curve (AUC). Note that an AUC value of 0 means the detection is always false, whereas an AUC value of 1 means a perfect detection. An AUC value of 0.5 represents random guessing, which is reflected by the diagonal line between (0, 0) and (1, 1) in the ROC curve.

The method described in [9] is used as the baseline algorithm for comparative purposes. Baseline #1 is fully implemented ac-

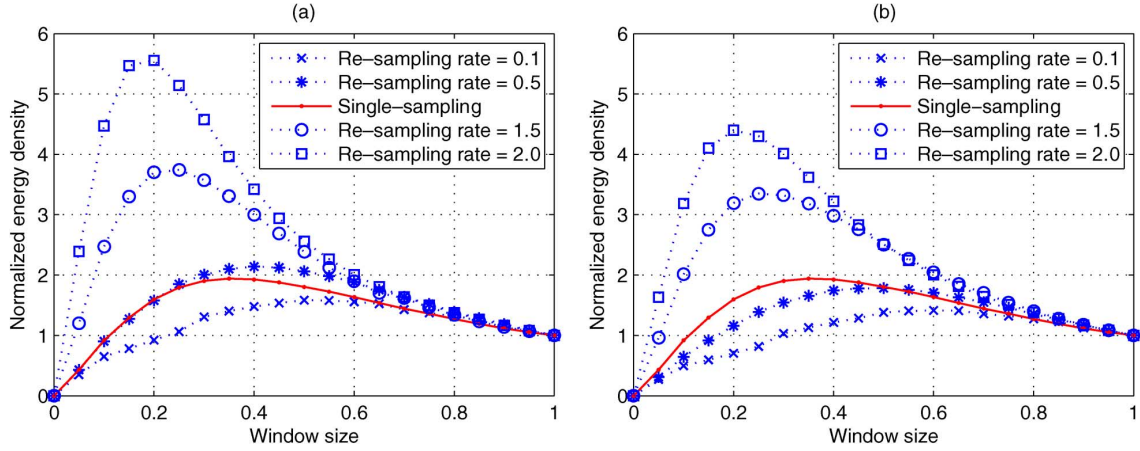


Fig. 6. Normalized energy density curves for a set of resampling rates, averaged over 7500 images from the BOSS database (after second-derivative high-pass filtering, with anti-aliasing, fixed content cropping). (a) Bilinear interpolation. (b) Bicubic interpolation.

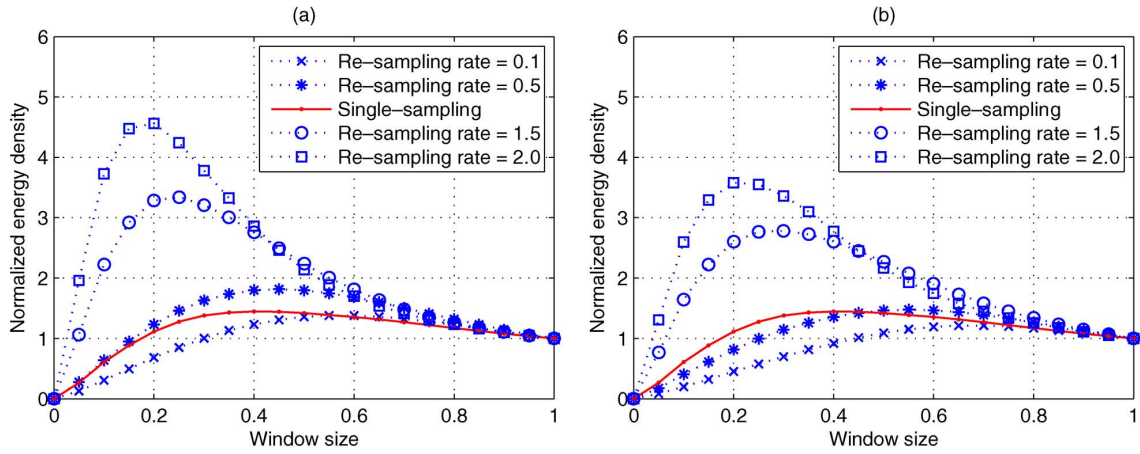


Fig. 7. Normalized energy density curves for a set of resampling rates, averaged over 7500 images from the BOSS database (after second-derivative high-pass filtering, with anti-aliasing, variable content cropping). (a) Bilinear interpolation. (b) Bicubic interpolation.

cording to [9]. In baseline #2, the discriminating features are implemented according to [9], but detection is based on the SVM classifier that is the same as our method. The purpose of baseline #2 is to help identify if the performance variation is due to different discriminating features or different detectors.

A. Practical Considerations About Resizing

A typical raw camera image in the BOSS database has the resolution of 5234×3487 . It is not necessary to perform the forensic analysis on the whole image. Instead, an image of smaller dimensions can be used. We therefore crop the original image by selecting its 512×512 central part. However, we can perform this cropping before or after resampling, as discussed next.

1) *Fixed Content*: With this first cropping option, the content of a single-sampled image and its corresponding resampled versions are the same. More specifically, suppose we have an original raw camera image with the resolution of 5234×3487 , the single-sampled version will be the central 512×512 part cropped from the original raw camera image. Based on this single-sampled version, a set of resampled images at different resampling rates are generated accordingly, either using bilinear interpolation or bicubic interpolation. As a result, we obtain a set of images at different resampling rates with the same image

content. Note that the dimensions of the resulting images at different resampling rates are different in this case.

2) *Variable Content*: With this second cropping option, the content of a single-sampled image and its corresponding resampled versions are different. Specifically, suppose we have an original raw camera image with the resolution of 5234×3487 . Then we first generate a set of resampled images at different resampling rates either using bilinear interpolation or bicubic interpolation. Both the original raw camera image and the resampled images are then cropped to 512×512 by selecting their central part. As a result, we have a set of images at different resampling rates with the same image size. Note that the image content at different resampling rates are different with this option.

Fig. 6 shows the normalized energy density curves with fixed content cropping, while Fig. 7 shows the equivalent curves with variable content cropping. This pair of figures suggests that the normalized energy density curves at different resampling rates remain well separated in both cases. However, the curves for “fixed content” cropping are more separable. In the remainder of this paper, we report results using “fixed content” cropping.

B. One Classifier Per Resampling Rate

In the first experiment, one SVM-based classifier is trained for each individual resampling rate ρ , which ranges from 0.1 to

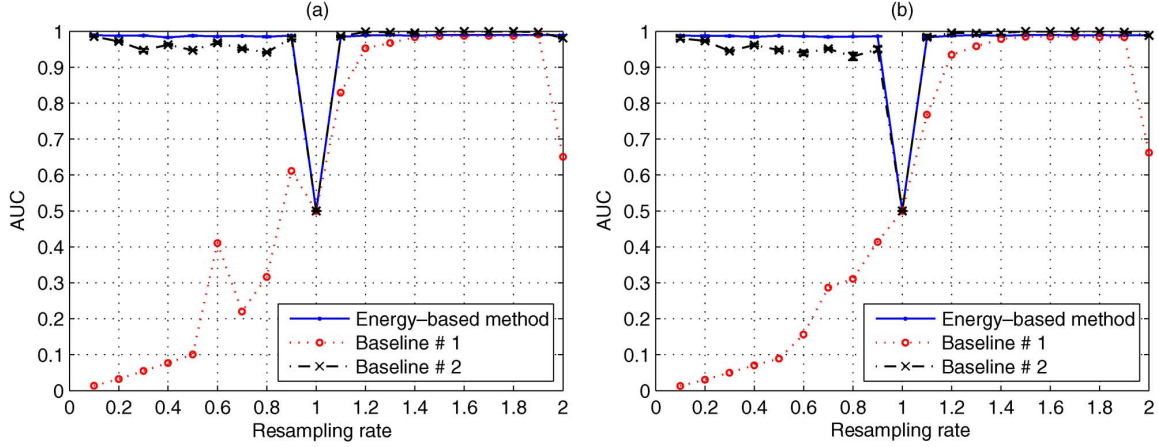


Fig. 8. Comparison of the energy method and the baseline, where one SVM-based classifier is trained for each resampling rate. The error bars indicate the standard deviations after 10 trials. (a) Bilinear interpolation. (b) Bicubic interpolation.

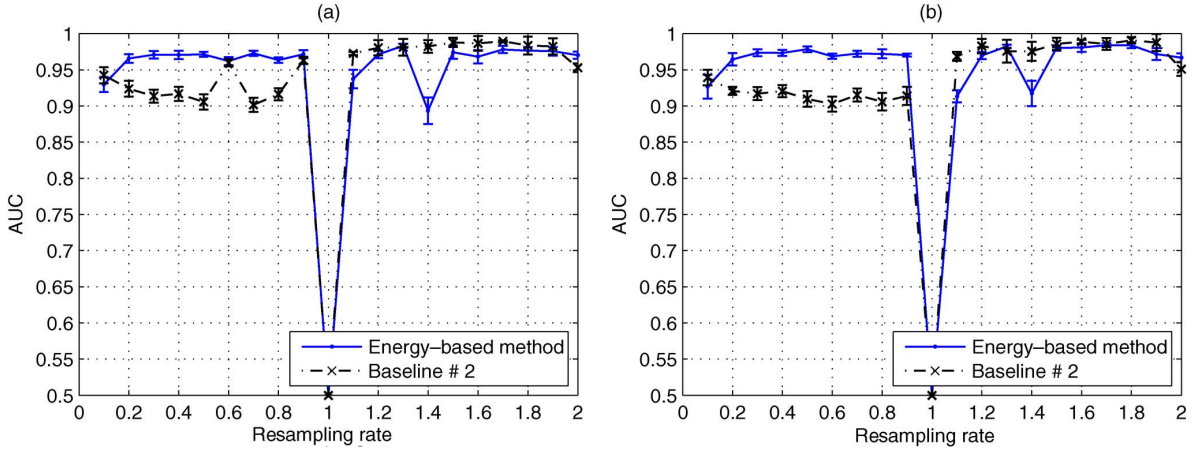


Fig. 9. Comparison of the energy method and the baseline, where only one SVM-based classifier is trained for all resampling rates. The error bars indicate the standard deviations after 10 trials. (a) Bilinear interpolation. (b) Bicubic interpolation.

2 in steps of 0.1, resulting in a total of 19 classifiers. The detection results are reported in Fig. 8 for both bilinear and bicubic interpolations. Note that during testing, the resampled image is only presented to its corresponding classifier.

In general, the detection performance of our method is comparable to that of the baseline algorithm described in [9] when resampling rates are greater than 1 (i.e., the resampled image is larger than the single-sampled one). Note that for this case, our method never reaches perfect detection, whereas the baseline #2 does. In contrast, when resampling rates are smaller than 1, our method performs better than the baseline algorithms. Note that the baseline #1 actually performs worse than random, i.e., AUC values are less than 0.5. This implies that the baseline #1 can discriminate between single-sampling and resampling, but that its decision is opposite to the truth. This characteristic was also observed in [20], where it was suggested that a two-sided hypothesis test is actually needed. For baseline #2, there is no such problem. Still, for down-sampling, the normalized energy density approach is almost always superior. In addition, we note that the error bars for all resampling rates are small, indicating that the proposed method is stable across all resampling rates. There is little difference in performance when either bilinear or bicubic interpolation is used.

C. Single Classifier for All Resampling Rates

We now consider a single SVM-based classifier for all resampling rates. This is a more realistic scenario where there is no prior knowledge of the resampling rate. If an image is used for training (i.e., within the 20% subset of the database), its single-sampled version and only one of its resampled versions are fed into the SVM-based classifier. This guarantees that the numbers of single-sampled and resampled images used to train the SVM-based classifier are the same. In addition, during training, we use the same number of resampled images for each of the 19 resampling rates.

As expected, the absolute detection performance of both our method and the baseline algorithm #2 declines slightly, as shown in Fig. 9. However, the degradation in performance is small. It is also noted that the relative performance of the two algorithms remains qualitatively the same. Once again, the error bars for all resampling rates are small, demonstrating the stability of the proposed algorithm. Note that 1) we do not show baseline #1 as this curve is identical to that in Fig. 8 since there is no training and 2) the scale of vertical axis in Fig. 9 is different from that in Fig. 8.

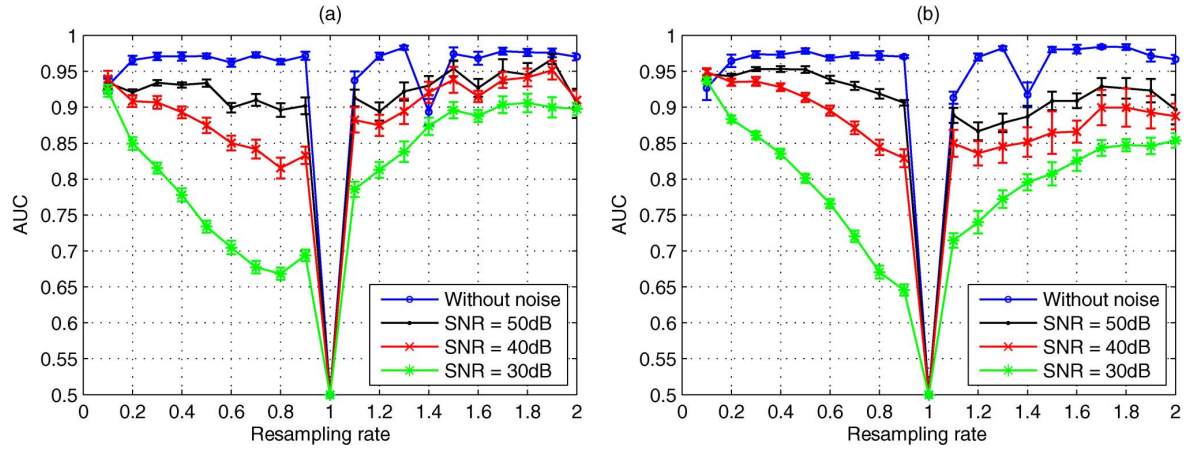


Fig. 10. Detection performance of resampling with different levels of noise based on the energy method, where only one SVM-based classifier is trained for all resampling rates, per SNR level. The error bars indicate the standard deviations after 10 trials. (a) Bilinear interpolation; (b) Bicubic interpolation.

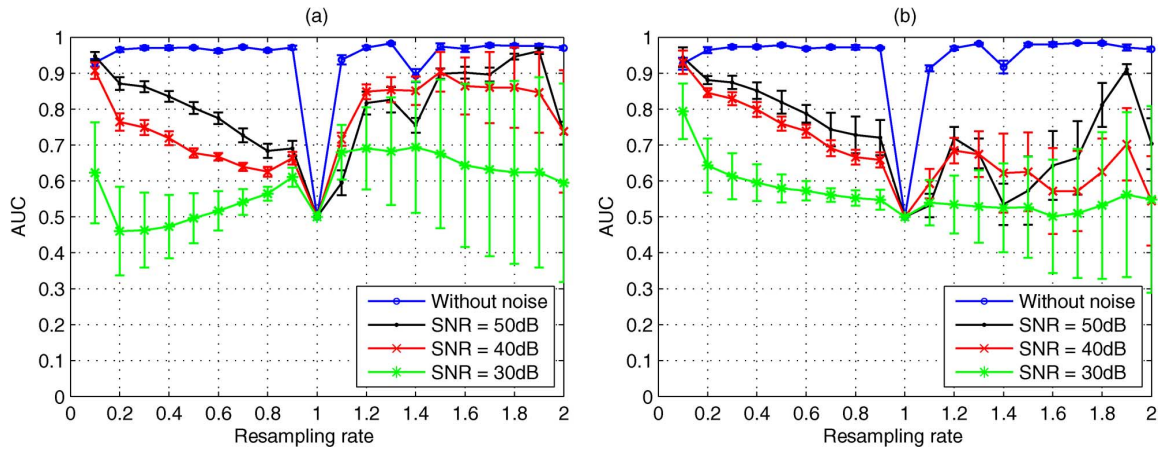


Fig. 11. Detection performance of resampling with different levels of noise based on the energy method, where only one SVM-based classifier is trained for all resampling rates, based on images without noise added. The error bars indicate the standard deviations after 10 trials. (a) Bilinear interpolation. (b) Bicubic interpolation.

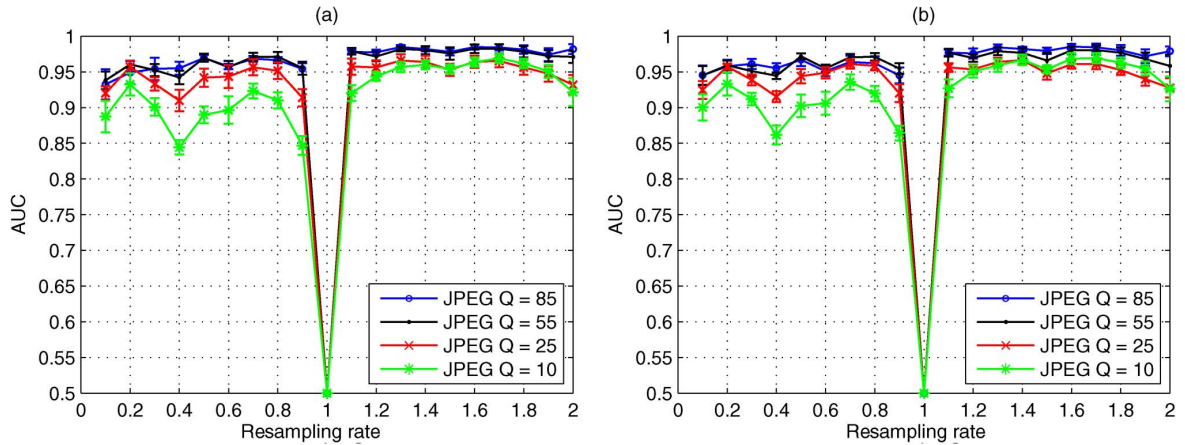


Fig. 12. Detection performance after JPEG compression for a set of different quality factors based on the energy method, where only one SVM-based classifier is trained for all resampling rates, per JPEG quality factor. The error bars indicate the standard deviations after 10 trials. (a) Bilinear interpolation. (b) Bicubic interpolation.

D. Detection of Resampling With Noise

To consider the effect of noise, we added white Gaussian noise to images *after* the resampling process, with resulting signal-to-noise ratios (SNR) of 1) 50 dB, 2) 40 dB, and 3) 30 dB. We note that noise is typically not visible for SNR greater than

45 dB. Image quality is usually considered acceptable for SNR between 36–45 dB, while image quality is considered poor for SNR below 36 dB.

The experimental results are reported in Fig. 10. Note that for each SNR value, we trained a single classifier for all resampling rates, with training data that had the corresponding noise

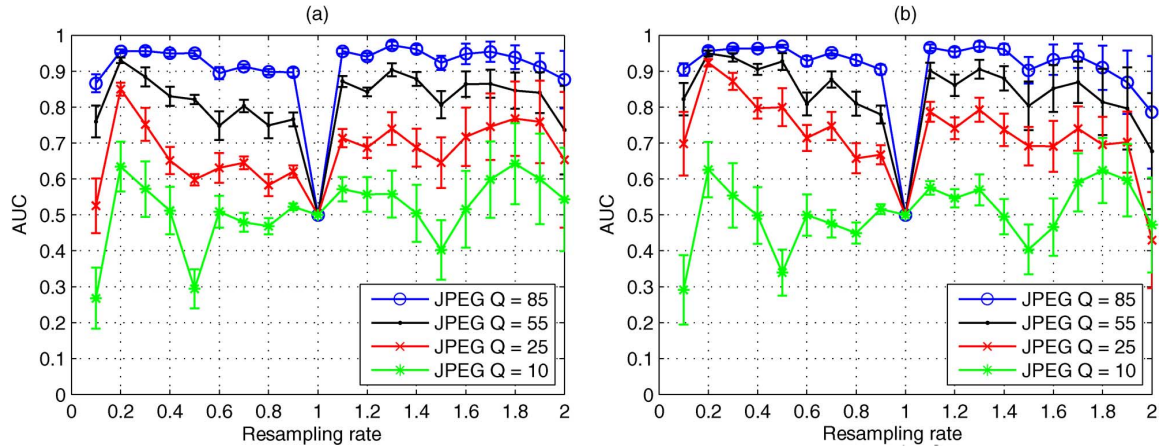


Fig. 13. Detection performance after JPEG compression for a set of different quality factors based on the energy method, where only one SVM-based classifier is trained for all resampling rates, based on images without JPEG compression. The error bars indicate the standard deviations after 10 trials. (a) Bilinear interpolation. (b) Bicubic interpolation.

level added. During testing, only the corresponding classifier was used for detection. As expected, performance degrades as the level of noise increases.

Fig. 11 reports experimental results with only a single classifier that is trained for all resampling rates, with no noise added to the training images. Clearly, the degradation in performance with regard to discriminating single-sampled and resampled noisy images is substantially worse when using the classifier trained by images without noise added.

E. Detection After JPEG Compression

In practice, an image may also be JPEG compressed *after* the resampling process [21]–[23]. We considered two scenarios. In the first scenario, we assume the JPEG quality factor is known. In this case, we train a single classifier for all resampling rates, per a given quality factor. In the second scenario, we assume no knowledge of the JPEG compression and use a classifier trained with images without JPEG compression.

The results are presented in Figs. 12 and 13 for JPEG compression with quality factors 10, 25, 55, and 85, where JPEG compression was performed using the MATLAB `imwrite()` function. As expected, the detection performance degrades with decreasing JPEG quality factors. However, when the quality factor is known, the performance degradation is relatively small, even for the JPEG quality factor of 10. However, the degradation in performance is substantially worse when no knowledge of the JPEG compression is available.

V. CONCLUSIONS

This paper investigated a new method for detecting resampling. It is based on computing the normalized energy density for various window sizes in the frequency domain of the second-derivative of the image. The energy density as a function of window size usually exhibits a peak whose location varies depending on the resampling rate. This characteristic curve is exploited to extract a 19-D feature vector that is subsequently used as input to a SVM-based classifier.

Experimental results on the BOSS database indicated that it is feasible to achieve good resampling detection performance even if a single classifier is trained for all resampling rates. The proposed algorithm exhibits similar detection performance compared to prior work [9] for up-sampling and significantly better

performance for down-sampling. Experimental results were reported on the average of 10 trials, where 20% of the database was used for training and 80% for testing. The variation across 10 trials was reported by error bars, and demonstrated good stability.

Further experiments examined the performance of the system when the resampled imagery is subsequently JPEG compressed or has noise added to it. If the JPEG quality factor is known, then a corresponding classifier can be trained. In this case, the degradation in performance is relatively small, even for a JPEG quality factor as low as 10. However, if no knowledge of the JPEG compression is available, the performance degradation is significantly worse. Similar results were obtained for the addition of white Gaussian noise.

A number of further investigations are possible. In particular, it would be interesting to consider the addition of noise, JPEG compression, and other forms of processing occurring *before* the resampling process. Also, given the significantly better performance of the classifier when the knowledge of quality factor is available, it would be interesting to examine how one might estimate the quality factor, and for example, the associated quantization table, and investigate the sensitivity of the classifier to errors in these estimates.

A further line of investigation would be to consider how the performance of the algorithm is affected when training and testing are based on images drawn from completely different databases. Further experiments could also be performed to investigate performance when “variable content” cropping is used.

Lastly, rather than just detect the presence of resampling, it would be interesting to estimate the resampling rate. The normalized energy density curves suggest that this may be possible.

REFERENCES

- [1] H. Farid, “Image forgery detection,” *IEEE Signal Process. Mag.*, vol. 26, no. 2, pp. 16–26, Mar. 2009.
- [2] I. J. Cox, M. L. Miller, J. A. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*, 2nd ed. San Mateo, CA: Morgan Kaufmann, 2008.
- [3] A. C. Popescu and H. Farid, “Statistical tools for digital forensics,” in *Proc. 6th Inf. Hiding Workshop, Lecture Notes Comput. Sci.*, 2004, vol. 3200, pp. 128–147.

- [4] W. Wang and H. Farid, "Exposing digital forgeries in video by detecting double MPEG compression," in *Proc. 8th ACM Workshop Multimedia Security*, 2006, pp. 37–47.
- [5] J. Lukáš and J. Fridrich, "Estimation of primary quantization matrix in double compressed JPEG images," in *Proc. Digital Forensic Res. Workshop*, 2003.
- [6] T. Pevný and J. Fridrich, "Detection of double-compression for applications in steganography," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 2, pp. 247–258, Jun. 2008.
- [7] X. Feng and G. Doërr, "JPEG recompression detection," in *Proc. SPIE*, 2010, vol. 7541, no. 20, pp. 1–12.
- [8] A. C. Popescu and H. Farid, "Exposing digital forgeries by detecting traces of resampling," *IEEE Trans. Signal Process.*, vol. 53, no. 2, pp. 758–767, Feb. 2005.
- [9] B. Mahdian and S. Saic, "Blind authentication using periodic properties of interpolation," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 3, pp. 529–538, Sep. 2008.
- [10] M. Kirchner, "Fast and reliable resampling detection by spectral analysis of fixed linear predictor residue," in *Proc. 10th ACM Workshop Multimedia Security*, 2008, pp. 11–20.
- [11] M. Kirchner, "Linear row and column predictors for the analysis of resized images," in *Proc. 12th ACM Workshop Multimedia Security*, 2010, pp. 13–18.
- [12] N. Dalgaard, C. Mosquera, and F. Pérez-González, "On the role of differentiation for resampling detection," in *Proc. 17th Int. Conf. Image Process.*, 2010, pp. 1753–1756.
- [13] D. Vazquez-Padin and F. Pérez-González, "Prefilter design for forensic resampling estimation," in *Proc. IEEE Int. Workshop Inf. Forensics Security*, 2011, pp. 1–6.
- [14] X. Feng, I. J. Cox, and G. Doërr, "An energy-based method for the forensic detection of resampled images," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2011, pp. 1–6.
- [15] X. Chen, J. Yang, J. Zhang, and A. Waibel, "Automatic detection of signs with affine transformation," in *Proc. 6th IEEE Workshop Appl. Comput. Vis.*, 2002, pp. 32–36.
- [16] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, "A comparison of affine region detectors," *Int. J. Comput. Vis.*, vol. 65, no. 1–2, pp. 43–72, 2005 [Online]. Available: <http://www.springerlink.com/content/744376233t370532/?MUD=MP>
- [17] P. Bas, T. Filler, and T. Pevný, "Break our steganographic system: The ins and outs of detecting BOSS," in *Proc. 13th Int. Workshop Inf. Hiding*, pp. 59–70 [Online]. Available: <http://boss.gipsa-lab.grenoble-inp.fr/>; Lecture notes in computer science
- [18] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York: Wiley-Interscience, 2000.
- [19] T. Fawcett, "ROC graphs: Notes and practical considerations for researchers," HP Laboratories, Palo Alto, CA, 2003.
- [20] F. Ucccheddu, A. De Rosa, A. Piva, and M. Barni, "Detection of resampled images: Performance analysis and practical challenges," in *Proc. 18th Eur. Signal Process. Conf.*, 2010, pp. 24–27.
- [21] A. Gallagher, "Detection of linear and cubic interpolation in JPEG compressed images," in *Proc. 2nd Canadian Conf. Comput. Robot. Vis.*, 2005, pp. 65–72.
- [22] M. Kirchner and T. Gloe, "On resampling detection in recompressed images," in *Proc. 1st IEEE Int. Workshop Inf. Forensics Security*, 2009, pp. 21–25.
- [23] Q. Liu and A. Sung, "A new approach for JPEG resize and image splicing detection," in *Proc. First ACM Workshop Multimedia Forensics*, 2009, pp. 43–48.



Xiaoying Feng (S'11–M'12) received the Ph.D. degree in computer science from the University College London, London, U.K., in 2012.

Before the Ph.D. research, she was a Software Engineer for the full life-cycle development of telecommunication projects. She specializes in software design and development. Her research interest is in the broad area of multimedia signal processing, specifically multimedia forensics.

Dr. Feng was the recipient of a Dorothy Hodgkin Postgraduate Award (DHPA) for her Ph.D. research, which was cofunded by Engineering and Physical Sciences Research Council (EPSRC) and British Telecommunications plc (BT).



Ingemar J. Cox (S'79–M'83–SM'95–F'06) received the B.Sc. from the University College London, London, U.K., in 1980, and the Ph.D. from Oxford University, Oxford, U.K., in 1983.

He was a member of the Technical Staff at AT&T Bell Labs, Murray Hill, NJ, from 1984 to 1989, where his research interests were focused on mobile robots. In 1989 he joined the NEC Research Institute, Princeton, NJ, as a Senior Research Scientist in the computer science division. At NEC, his research shifted to problems in computer vision and he was

responsible for creating the computer vision group at NECI. He is currently Professor and Director of Research in the Department of Computer Science, University College London, where he is Head of the Future Media Group. He has worked on problems to do with stereo and motion correspondence and multimedia issues of image database retrieval and watermarking. From 1997 to 1999, he served as Chief Technical Officer of Signafy, Inc, a subsidiary of NEC responsible for the commercialization of watermarking. Between 1996 and 1999, he led the design of NEC's watermarking proposal for DVD video disks and later collaborated with IBM in developing the technology behind the joint "Galaxy" proposal supported by Hitachi, IBM, NEC, Pioneer and Sony. In 1999, he returned to the NEC Research Institute as a Research Fellow. He is coauthor of a book entitled "Digital Watermarking" (Morgan Kaufmann, 2002) and its second edition "Digital Watermarking and Steganography" (Morgan Kaufmann, 2008), and the co-editor of "Autonomous Robots Vehicles" (Springer, 1990) and "Partitioning Data Sets: With Applications to Psychology, Computer Vision and Target Tracking" (American Mathematical Society, 1995).

Dr. Cox is a Fellow of the IET (formerly IEE) and the British Computer Society, and a member of the U.K. Computing Research Committee. He was founding Co-Editor in chief of the IEE Proceedings on Information Security and is an Associate Editor of the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY. He was the recipient of a Royal Society Wolfson Fellowship (2002–2007). In 1999, he was awarded the IEEE Signal Processing Society Best Paper Award (Image and Multidimensional Signal Processing Area) for a paper he coauthored on watermarking.



Gwenaél Doërr (M'06) received the M.Sc. degree in telecommunications systems from Télécom Sud-Paris, Evry, France, in 2001, and the Ph.D. degree in signal and image processing from the Université de Nice Sophia-Antipolis, Nice, France, in 2005.

From August 2005 to December 2009, he was a Lecturer on digital rights management at the Department of Computer Science of University College London, London, U.K. In Spring 2008, he was a Visiting Scholar at HP Labs, Palo Alto, CA, to work on the interoperability of DRM systems with

Ton Kalker. In January 2010, he joined the Security & Content Protection Labs of Technicolor R&D France, Cesson-Sévigné, France, as a Senior Research Scientist on content protection. His research interests mostly relate to signal processing for multimedia content protection and security. In particular, his recent works focused on digital watermarking, passive forensics for multimedia, content protection systems, and content fingerprinting. As part of his work, he has been involved in a number of collaborative project e.g. the European project *Certimark* and in the European Network of Excellence *E-Crypt*.

Dr. Doërr is a member of the IEEE's Signal Processing Society Technical Committee on Information Forensics and Security. He is currently a Distinguished Member of Technicolor's Fellow Network. He is an Associate Editor for the IEEE SIGNAL PROCESSING LETTERS, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, the IET Proceedings on Information Security, and the EURASIP Journal on Information Security. He coorganized IH'07 in St Malo, France and IEEE WIFS'09 in London, U.K.