Anti-Forensics of Digital Image Compression

Matthew C. Stamm, *Student Member, IEEE*, and K. J. Ray Liu, *Fellow, IEEE*

***Abstract*—As society has become increasingly reliant upon digital images to communicate visual information, a number of forensic techniques have been developed to verify the authenticity of digital images. Amongst the most successful of these are techniques that make use of an image's compression history and its associated compression fingerprints. Little consideration has been given, however, to *anti-forensic* techniques capable of fooling forensic algorithms. In this paper, we present a set of anti-forensic techniques designed to remove forensically significant indicators of compression from an image. We do this by first developing a generalized framework for the design of anti-forensic techniques to remove compression fingerprints from an image's transform coefficients. This framework operates by estimating the distribution of an image's transform coefficients before compression, then adding *anti-forensic dither* to the transform coefficients of a compressed image so that their distribution matches the estimated one. We then use this framework to develop anti-forensic techniques specifically targeted at erasing compression fingerprints left by both JPEG and wavelet-based coders. Additionally, we propose a technique to remove statistical traces of the blocking artifacts left by image compression algorithms that divide an image into segments during processing. Through a series of experiments, we demonstrate that our anti-forensic techniques are capable of removing forensically detectable traces of image compression without significantly impacting an image's visual quality. Furthermore, we show how these techniques can be used to render several forms of image tampering such as double JPEG compression, cut-and-paste image forgery, and image origin falsification undetectable through compression-history-based forensic means.**

***Index Terms*—Anti-forensics, anti-forensic dither, digital forensics, image compression, JPEG compression.**

I. Introduction

DUE TO the widespread availability of digital cameras and the rise of the Internet as a means of communication, digital images have become an important method of conveying visual information. Unfortunately, the ease with which digital images can be manipulated by photoediting software has created an environment where the authenticity of digital images is often in doubt. To prevent digital image forgeries from being passed off as unaltered originals, researchers have developed a variety of digital image forensic techniques. These techniques are designed to determine an image's originating camera [1], trace its processing history [2], and determine its authenticity [3], [4], all without relying on an extrinsically inserted watermark or access to the original image. Instead, these techniques make use of *intrinsic fingerprints* introduced into an image by editing operations or the image formation process itself [5].

Image compression fingerprints are of particular forensic significance due the fact that most digital images are subjected to compression either by the camera used to capture them, during image storage, or for the purposes of digital transmission over the Internet. Techniques have been developed to determine if an image saved in a lossless format has ever undergone JPEG compression [6], [7] or other types of image compression including wavelet-based techniques [7]. If evidence of JPEG compression is detected, the quantization table used during compression can be estimated [6]. Because most digital cameras and image editing software use proprietary JPEG quantization tables when compressing an image, an image's origin can be identified by matching the quantization tables used to compress the image with those in a database of quantization table and camera or software pairings [8]. If the quantization tables are matched with those used by image editing software, the authenticity of the image can be called into question. Recompressing an image which has previously been JPEG compressed, also known as double JPEG compression, can be detected [9], [10] and the quantization table used during the initial application of JPEG compression can be estimated. Localized evidence of double JPEG compression can be used to identify image forgeries [11] as well as localized mismatches in an image's JPEG block artifact grid [12].

Though many existing forensic techniques are capable of detecting a variety of standard image manipulations, they do not account for the possibility that *anti-forensic* operations may be designed and used to hide image manipulation fingerprints. This is particularly important because it calls into question the validity of forensic results indicating the absence of image tampering. It may be possible for an image forger familiar with signal processing to secretly develop anti-forensic operations and use them to create undetectable image forgeries. As a result, several existing forensic techniques may contain unknown vulnerabilities.

In order to combat the creation and spread of undetectable image forgeries, it is necessary for image forensics researchers themselves to develop and study anti-forensic operations. By doing so, researchers can be made aware of which forensic techniques are capable of being deceived, thus preventing altered images from being represented as authentic and allowing forensic examiners to establish a degree of confidence in their findings. Furthermore, it is likely that many anti-forensic operations will leave behind detectable fingerprints of their own. If these fingerprints can be discovered, forensic techniques can be designed to detect the use of anti-forensic operations. It is also possible that anti-forensic operations may be used to

Manuscript received September 10, 2010; revised December 23, 2010; accepted February 03, 2011. Date of publication February 24, 2011; date of current version August 17, 2011. This work was supported in part by AFOSR Grant FA95500910179. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Mauro Barni.

The authors are with the Department of Electrical and Computer Engineering, University of Maryland, College Park, MD 20742 USA (e-mail: mcstamm@umd.edu; kjrliu@umd.edu).

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

Digital Object Identifier 10.1109/TIFS.2011.2119314

provide intellectual property protection. This would be done by integrating them into digital image and video cameras to prevent the reverse engineering of proprietary signal processing components through digital forensic means.

At present, very little anti-forensics research has been published. To the best of our knowledge, the only prior work studying digital image anti-forensics are techniques to remove traces of image resizing and rotation [13], forge the photoresponse nonuniformity noise fingerprint left in an image by a digital camera's electronic sensor [14], and to artificially synthesize color filter array artifacts [15]. In this paper, we present a set of anti-forensic operations capable of removing compression fingerprints from digital images. Since most modern lossy image compression techniques involve transform coding, we propose a framework for the removal of quantization fingerprints from a compressed image's transform coefficients by adding *anti-forensic dither* to them. We use this framework to develop anti-forensic operations to remove quantization artifacts from the discrete cosine transform (DCT) coefficients of JPEG compressed images and from the wavelet coefficients of wavelet-based schemes such as JPEG 2000, SPIHT, and EZW [16], [17]. Additionally, we propose an anti-forensic operation to remove statistical traces of blocking artifacts from JPEG compressed images. We then experimentally demonstrate that our proposed anti-forensic operations can be used to fool a variety of compression fingerprint-based forensic algorithms designed to detect single and double JPEG compression, wavelet-based image compression, determine an image's origin, and detect cut-and-paste image forgeries [18].

The organization of this paper is as follows. In Section II, we discuss the quantization fingerprints left by image transform coders and propose a generalized framework for their removal. We adapt this framework for use with JPEG compression in Section III and wavelet-based compression in Section IV. In Section V, we propose an anti-forensic technique capable of removing statistical traces of blocking artifacts. We present the results of several experiments designed to evaluate the performance of each of our proposed anti-forensic techniques in Section VI. In Section VII, we discuss how these techniques can be used to render certain forms of image tampering such as double JPEG compression, cut-and-paste image forgery, and image origin falsification undetectable through compression history-based forensic means. Finally, we conclude this paper in Section VIII.

II. Anti-Forensic Framework

Virtually all modern lossy image compression techniques are subband coders, which are themselves a subset of transform coders. Transform coders operate by applying a mathematical transform to a signal, then compressing the transform coefficients. Subband coders are transform coders that decompose the signal into different frequency bands or subbands of transform coefficients. Typical lossy image compression techniques operate by applying a two-dimensional invertible transform, such as the DCT or discrete wavelet transform (DWT), to an image as a whole, or to each set of pixels within an image that has been segmented into a series of disjoint sets. As a result, the image or set of pixels is mapped into multiple subbands of transform coefficients, where each transform coefficient is denoted $X \in \mathbb{R}$.

Once obtained, each transform coefficients must be mapped to a binary value both for storage and to achieve lossy compression. This is achieved through the process of quantization, in which the binary representation $\hat{X}$ of the transform coefficient X is assigned the value $\hat{x}$ according to the equation

$$\hat{X} = \hat{x} \qquad \text{if } b_k \leq X < b_{k+1} \tag{1}$$

where b_k and b_{k+1} denote the boundaries of the quantization interval over which X maps to the value $\hat{x}$. Because some subbands of transform coefficients are less perceptually important than others, and thus can accommodate greater loss during the quantization process, the set of quantization interval boundaries is chosen differently for each subband. After each transform coefficient is given a binary representation, the binary values are reordered into a single bit stream which is often subjected to lossless compression.

When the image is decompressed, the binary bit stream is first rearranged into its original two-dimensional form. Each decompressed transform coefficient Y is assigned a value through dequantization. During this process, each binary value is mapped to a quantized transform coefficient value q belonging to the discrete set $\mathcal{Q} = \{\ldots, q_{-1}, q_0, q_1, \ldots\}$. Each dequantized transform coefficient value can be directly related to its corresponding original transform coefficient value by the equation

$$Y = q_k \qquad \text{if } b_k \leq X < b_{k+1}. \tag{2}$$

After dequantization, the inverse transform is performed on the set of transform coefficients and the resulting values are projected back into the set of allowable pixel values $\mathcal{P} = \{0, \ldots, 255\}$. If the image was segmented, this process is repeated for each segment and the decompressed segments are joined together to create the decompressed image; otherwise, this process reconstructs the decompressed image as a whole.

By performing image compression in this manner, a distinct fingerprint is introduced into the transform coefficients of an image. When examining an unaltered image's transform coefficient values within a particular subband, they will likely be distributed according to a smooth, continuous distribution. This is not the case for images which have undergone image compression, since the processes of quantization and dequantization force the transform coefficients of a compressed image to take values within the discrete set $\mathcal{Q}$. In practice, the act of projecting the decompressed pixel values perturbs the transform coefficient values, though the transform coefficients of a previously compressed image still cluster tightly around elements of $\mathcal{Q}$. These fingerprints, known as transform coefficient quantization artifacts, are used by the majority of compression artifact-based forensic techniques to identify single or double compression, determine an image's origin, or identify image forgeries. They can be clearly seen in Fig. 1, which shows the histogram of one subband of DCT coefficients from an image before and after JPEG compression.

If the image was divided into segments during compression, another compression fingerprint may arise. Because of the lossy

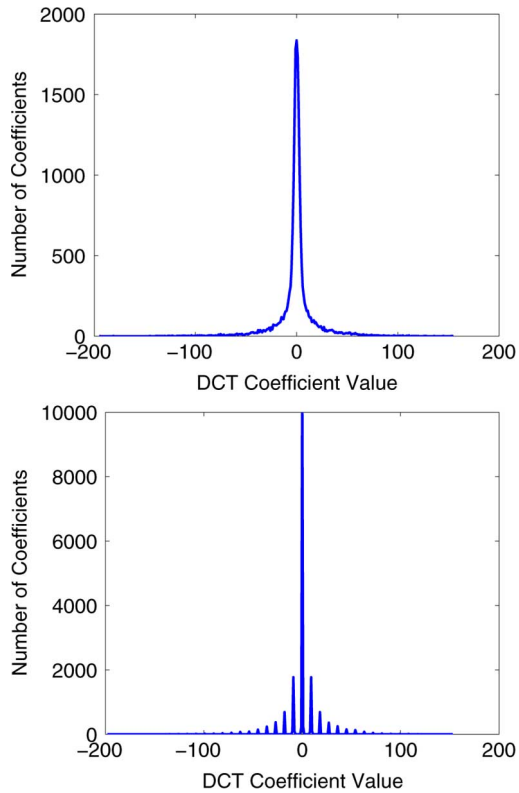

Fig. 1. Top: Histogram of DCT coefficients from an uncompressed image. Bottom: Histogram of DCT coefficients from the same image after JPEG compression.

nature of image transform coding, pixel domain discontinuities often arise across the boundaries of these segments. Research has shown that these discontinuities can be statistically detected even when they are not visible [6]. These discontinuities are known as blocking artifacts, since in the majority of cases the image segments take the form of square blocks. While important, these fingerprints are less frequently used by forensic algorithms, and their anti-forensic removal will be discussed in Section V.

To remove transform coefficient quantization artifacts from a compressed image we propose the following generalized framework. First, we model the distribution of the transform coefficients for a given subband prior to quantization using a parametric model $P(X = x) = f(x, \theta)$ with parameter θ. Next, we estimate the value of θ from the quantized transform coefficients. We then anti-forensically modify each quantized transform coefficient by adding specially designed noise, which we refer to as *anti-forensic dither*, to its value according to the equation

$$Z = Y + D \tag{3}$$

where D is the anti-forensic dither and Z is the anti-forensically modified coefficient. The distribution of the anti-forensic dither is chosen so that it corresponds to a renormalized and recentered segment of the model distribution for that subband, where the segment is centered at the quantized coefficient value and the segment's length is equal to the length of the quantization interval. Because the probability that the quantized coefficient value is q_k is given by

$$P(Y = q_k) = \int_{b_k}^{b_{k+1}} f(x, \theta) dx \tag{4}$$

the anti-forensic dither's distribution is given by the formula

$$P(D = d \,|\, Y = q_k) = \frac{f(q_k + d, \theta)}{\int_{b_k}^{b_{k+1}} f(x, \theta) dx} \mathbb{1}(b_k \leq q_k + d < b_{k+1}). \tag{5}$$

As a consequence of this, the anti-forensic dither distribution will be conditionally dependent not only upon the value of θ, but on the value of the coefficient to which the dither is to be added as well.

Choosing the anti-forensic dither distribution in this manner yields two main benefits; the anti-forensically modified coefficient distribution will theoretically match the transform coefficient distribution before quantization and an upper bound can be placed on the distance between each unquantized transform coefficient and its anti-forensically modified counterpart. To prove the first property, we make use of the fact that $P(Z = z \,|\, Y = q_k) = P(D = z - q_k \,|\, Y = q_k)$. We then use the law of total probability to write an expression for the anti-forensically modified coefficient distribution as shown in (6), at the bottom of the page, thus proving $P(Z = z) = P(X = z)$. This is important because it proves that forensic analysis of the transform coefficient distribution of an image should be unable to distinguish an unaltered image from an anti-forensically modified one, provided that the distribution of unmodified coefficients is modeled accurately and the parameter θ is correctly estimated.

An upper bound can be placed on the distance between an unquantized transform coefficient value and its anti-forensically modified counterpart by first examining the distance

$$\begin{aligned} P(Z = z) &= \sum_k P(Z = z \,|\, Y = q_k) P(Y = q_k) \\ &= \sum_k \left(\frac{f(q_k + (z - q_k), \theta)}{\int_{b_k}^{b_{k+1}} f(x, \theta) dx} \mathbb{1}(b_k \leq q_k + d < b_{k+1}) \right) \int_{b_k}^{b_{k+1}} f(x, \theta) dx \\ &= \sum_k f(z, \theta) \mathbb{1}(b_k \leq q_k + d < b_{k+1}) \\ &= f(z, \theta) \end{aligned} \tag{6}$$

an unquantized coefficient and its corresponding quantized value. Assuming that each quantized value lies at the center of its corresponding quantization interval, this distance can be trivially bounded as follows:

$$|X - Y| \leq \max_k \frac{1}{2}|b_k - b_{k+1}|. \tag{7}$$

Because each anti-forensically modified coefficient value must lie within the quantization interval encompassing the modified quantized coefficient value, the bound placed on $|X - Y|$ also holds for $|Y - Z|$. As a result, the distance between an unquantized and anti-forensically modified transform coefficient value is upper bounded by

$$|X - Z| \leq \max_k |b_k - b_{k+1}|. \tag{8}$$

If the transform coefficients are subjected to uniform quantization, i.e., $|b_k - b_{k+1}| = Q$ for all k, this bound can be rewritten as $|X - Z| \leq Q$. Though it is often difficult to analytically translate distortion introduced in the transform coefficient domain to the pixel domain, this upper bound demonstrates that the amount of distortion introduced into the image through anti-forensic modification is determined by the compression strength.

III. JPEG Anti-Forensics

In this section, we provide a brief overview of JPEG compression, then present our anti-forensic technique designed to remove compression fingerprints from a JPEG compressed image's DCT coefficients.

A. JPEG Compression Overview

For a gray-scale image, JPEG compression begins by segmenting an image into a series of nonoverlapping 8×8 pixel blocks, then computing the two-dimensional DCT of each block. The resulting transform coefficients are then quantized by dividing each coefficient value by its corresponding entry in predetermined quantization matrix Q, then rounding the resulting value to the nearest integer. Accordingly, a quantized DCT coefficient at the (i, j) block position is represented by the value $\hat{X} = \text{round}(X/Q_{i,j})$. Finally, the binary representations of each quantized DCT coefficient are reordered into a single bit stream using the zigzag scan order then losslessly encoded. Color images are compressed in a similar manner; however, they require additional preprocessing. First, the image is transformed from the RGB to the YCbCr color space. Next, the chrominance layers are typically down-sampled by a factor of two in both the horizontal and vertical directions. After this has been performed, compression continues as if each color layer were an independent gray-scale image.

A JPEG image is decompressed by first losslessy decoding the bit stream, then rearranging the integer representations of the quantized DCT coefficients back into their original 8×8 block form. Next, the DCT coefficient values are dequantized by multiplying the integer representation of each DCT coefficient value by its corresponding entry in the quantization matrix. The inverse DCT of each block of coefficients is computed and the resulting pixel values are projected back into the set $\mathcal{P}$ of allowable pixel values. The decompressed gray-scale image or color layer is then reassembled from the series decoded blocks. If a color image that was subject to chrominance layer down-sampling is decompressed, each of the down-sampled layers are returned to their original size through interpolation, then the image is transformed back into the RGB color space.

As was discussed in Section II, JPEG compression will result in two forensically significant fingerprints: DCT coefficient quantization fingerprints and blocking artifacts. DCT coefficient quantization fingerprints, which can be seen in the DCT coefficient histograms displayed in Fig. 1, correspond to the clustering of DCT coefficient values around integer multiples of their corresponding entry in the quantization. This occurs because a quantized DCT coefficient value Y is related to its unquantized counterpart by the equation $Y = Q_{i,j}\text{round}(X/Q_{i,j})$. JPEG blocking artifacts are the discontinuities which occur across the 8×8 pixel block boundaries that arise due to pixel value perturbations caused by DCT coefficient quantization. Anti-forensic removal of these fingerprints will be discussed in detail in Section V.

B. DCT Coefficient Quantization Fingerprint Removal

In accordance with the anti-forensic framework which we outlined in Section II, we begin by modeling the distribution of coefficient values within a particular ac DCT subband using the Laplace distribution [19]

$$P(X = x) = \frac{\lambda}{2}e^{-\lambda|x|}. \tag{9}$$

Though we use this model for each ac subband of the DCT, each subband will have its own unique value of λ. Using this model and the quantization rule described above, the coefficient values of an ac subband of DCT coefficients within a previously JPEG compressed image will be distributed according to the discrete Laplace distribution

$$P(Y = y) = \begin{cases} 1 - e^{-\lambda Q_{i,j}/2}, & \text{if } y = 0 \\ e^{-\lambda|y|}\sinh\left(\frac{\lambda Q_{i,j}}{2}\right), & \text{if } y = kQ_{i,j} \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

where $k \in \mathbb{Z}$, $k \neq 0$.

To anti-forensically modify a previously JPEG compressed image, we first perform the initial steps in JPEG compression (i.e., color space transformation, segmentation into blocks, DCT) to obtain a set of DCT coefficients from the image. Because the final stages of JPEG decompression involve projecting the decompressed pixel values back into $\mathcal{P}$, the DCT coefficient values obtained from the image will be perturbed from their quantized values. We assume these perturbations are small enough that they do not move a coefficient value into a different quantization interval. As a result, the quantized coefficient values can be obtained by repeating the quantization process upon the perturbed coefficients Y' so that $Y = Q_{i,j}\text{round}(Y'/Q_{i,j})$.

Next, we obtain a maximum likelihood estimate the model parameter λ independently for each ac subband of DCT coefficients using the quantized coefficients [20]. By doing this, we can use our model to obtain an estimate of each ac subband's coefficient distribution before JPEG compression. We define $N = N_0 + N_1$ as the total number of observations of the current DCT subband, N_0 as the number DCT subband of coefficients

taking the value zero, N_1 as the number of nonzero valued coefficients, and $S = \sum_{k=1}^{N} |y_k|$. The model parameter estimate, which we denote $\hat{\lambda}$, is calculated using the equation

$$\hat{\lambda} = -\frac{2}{Q_{i,j}} \ln(\gamma) \tag{11}$$

where γ is defined as

$$\gamma = \frac{-N_0 Q_{i,j}}{2NQ_{i,j} + 4S} + \frac{\sqrt{N_0^2 Q_{i,j}^2 - (2N_1 Q_{i,j} - 4S)(2NQ_{i,j} + 4S)}}{2NQ + 4S}. \tag{12}$$

After λ has been estimated, we add anti-forensic dither to each DCT coefficient in an ac subband. Because we model the coefficient distribution before quantization using the Laplace distribution, the expression for the anti-forensic dither's distribution given in (5) simplifies to one of two equations depending upon the magnitude of the quantized DCT coefficient value to which the dither is added. For zero-valued quantized coefficients, the anti-forensic dither distribution is chosen to be

$$P(D = d \,|\, Y = 0) = \begin{cases} \frac{1}{c_0} e^{-\hat{\lambda}|d|}, & \text{if } \frac{-Q_{i,j}}{2} \geq n > \frac{Q_{i,j}}{2} \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

where $c_0 = 1 - e^{-\hat{\lambda} Q_{i,j}/2}$. The distribution of the anti-forensic dither added to nonzero quantized DCT coefficients is

$$P(D = d \,|\, Y = q_k) = \begin{cases} \frac{1}{c_1} e^{-\mathrm{sgn}(q_k)\hat{\lambda}(d + Q_{i,j}/2)}, & \text{if } \frac{-Q_{i,j}}{2} \geq n > \frac{Q_{i,j}}{2} \\ 0, & \text{otherwise} \end{cases} \tag{14}$$

where $c_1 = (1/\hat{\lambda})(1 - e^{-\hat{\lambda} Q_{i,j}})$. An important benefit of the anti-forensic dither distributions taking these forms is that they reduce the complexity of generating the anti-forensic dither. Rather than drawing dither samples from a number of distributions equal to the number of distinct quantized DCT coefficient values within an ac subband, anti-forensic dither samples need only to be drawn from one of the two distributions displayed in (13) and (14).

As we demonstrated for the general case in (6), using these anti-forensic dither distributions will ensure that the distribution of anti-forensically modified DCT coefficients within an ac subband will match its modeled unquantized coefficient distribution. By using the expressions for the quantized coefficient distribution as well as the anti-forensic dither distribution given in (10), (13), and (14), and using the law of total probability we may write (15), shown at the bottom of the page.

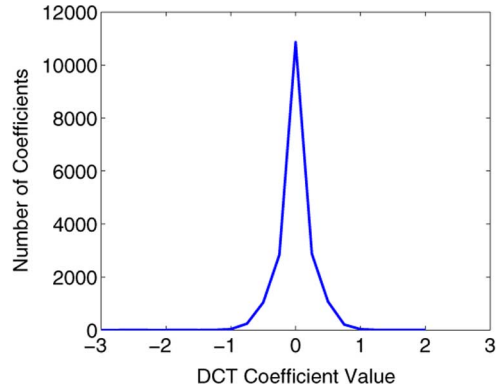

Fig. 2. Histogram of perturbed DCT coefficient values from a DCT subband in which all coefficients were quantized to zero during JPEG compression.

In most quantization tables, larger quantization step sizes are used for high-frequency DCT subbands because changes to these subbands are less perceptually significant. Furthermore, it has been observed that the variance of coefficient values within a DCT subband decreases as one moves from low-frequency to high-frequency subbands. Because of this, all of the coefficient values of certain high-frequency DCT subbands will be quantized to zero in some images during JPEG compression. Correspondingly, no estimate of λ can be obtained for these DCT subbands, rendering us unable to anti-forensically modify their coefficient values. Fortunately, the DCT coefficient value perturbations caused by the final steps in JPEG decompression result the coefficient values of these subbands taking on a plausible distribution, as can be seen in Fig. 2. As a result, we do not need to anti-forensically modify the coefficients of these DCT subbands.

Because the distribution of the dc subband of DCT coefficients varies greatly from image to image, no accurate parametric model for this distribution exists. Instead, we model the distribution of dc subband of unquantized DCT coefficients as being uniformly distributed within a quantization interval. As a consequence, we are able to create a set of anti-forensically modified coefficients whose distribution approximates

$$\begin{aligned} P(Z = z) &= \sum_k P(Z = z \,|\, Y = q_k) P(Y = q_k) \\ &= \sum_{q_k \neq 0} \frac{1}{c_1} e^{-\mathrm{sgn}(q_k)\hat{\lambda}(z - q_k + Q_{i,j}/2)} e^{-\hat{\lambda}|q_k|} \sinh\left(\frac{\hat{\lambda} Q_{i,j}}{2}\right) \mathbb{1}\left(|z - q_k| \leq \frac{Q_{i,j}}{2}\right) \\ &\quad + \frac{1}{c_0} e^{-\hat{\lambda}|z|} (1 - e^{-\hat{\lambda} Q_{i,j}/2}) \mathbb{1}\left(|z| \leq \frac{Q_{i,j}}{2}\right) \\ &= \frac{\hat{\lambda}}{2} e^{-\hat{\lambda}|z|} \end{aligned} \tag{15}$$

Fig. 3. Chrominance layer reconstruction interleaving pattern.

the unquantized distribution by adding uniformly distributed anti-forensic dither to the quantized dc subband of DCT coefficients. The dither is chosen to be zero mean over a support interval equal in length to the quantization interval so that

$$P(D=d)=\begin{cases}\frac{1}{Q_{i,j}}, & \text{if } \frac{-Q_{i,j}}{2}\le n<\frac{Q_{i,j}}{2}\\ 0, & \text{otherwise.}\end{cases} \tag{16}$$

Though this could in theory introduce step discontinuities into the distribution of the dc subband of anti-forensically modified DCT coefficients, we have experimentally observed that this is rarely the case. The absence of step discontinuities from the empirical distribution of anti-forensically modified coefficients is likely due to the fact that the dynamic range of dc DCT values is typically sufficiently large in comparison to the quantization interval that relatively few dc coefficients are quantized to any given value. As a result, too few anti-forensically modified coefficient values exist over an interval for step discontinuities to be discernible.

After the anti-forensically modified DCT coefficients are obtained, the inverse DCT of each block of coefficients is performed and the resulting blocks of pixel values are assembled into the anti-forensically modified image. If a color image subjected to chrominance layer down-sampling during JPEG compression undergoes anti-forensic modification, a number equal to the down-sampling factor of independently generated anti-forensically modified versions of each down-sampled chrominance layer is created. Each independent version of the anti-forensically modified down-sampled chrominance layer is then interleaved to create one equal in size to the full sized image. For images that undergo chrominance layer down-sampling by a factor of two in each direction as is most commonly the case, the anti-forensically modified down-sampled layers are interleaved using the pattern shown in Fig. 3.

IV. Wavelet-Based Image Compression Anti-Forensics

In this section, we begin by providing a brief overview of several wavelet-based image compression techniques and their forensically significant compression fingerprints. After this, we present our anti-forensic technique designed to remove compression fingerprints from the wavelet coefficients of an image compressed using a wavelet-based technique.

A. Wavelet-Based Compression Overview

Though several wavelet-based image compression techniques exist such as SPIHT, EZW, and most popularly JPEG 2000, they all operate in a similar fashion and leave behind similar compression fingerprints. Techniques such as JPEG 2000 begin compression by first segmenting an image into fixed sized nonoverlapping rectangular blocks known as "tiles," while others operate on the image as a whole. Next, the two-dimensional DWT of the image or each image tile is computed, resulting in four subbands of wavelet coefficients. Because these subbands correspond to either high (H) or low (L) frequency DWT coefficients in each spatial dimension, the four subbands are referred to using the notation LL, LH, HL, and HH. The DWT of the LL subband is computed an additional $M-1$ times, resulting in an M-level wavelet decomposition of the image or tile.

After this, tree-based compression techniques such as SPIHT or EZW divide the set of DWT into separate bit planes which are each processed independently. Within each bit plane, a tree-like structure known as a significance map is constructed detailing the locations of nonzero coefficients at that bit level [21]. Because the locations of zero-valued bit plane coefficients are correlated across DWT subbands, this allows for the efficient storage of each bit plane. The significance maps of each bit plane are then reordered into a single bit stream, with the map of the most significant bit plane occurring at the beginning of the bit stream, then proceeding in descending order of significance. To achieve lossy compression, the bit stream is truncated to a fixed number of bits according to a predefined bit budget.

JPEG 2000 achieves lossy compression in an alternate manner [22]. First, each subband of DWT coefficients are independently quantized using their own fixed quantization step size. Next, the binary representations of the quantized coefficients are divided into bit planes and separated into code blocks which are then entropy coded and reordered into a single bit stream. Because compression is achieved through quantization, the bit stream does not undergo truncation.

Image decompression begins by obtaining the set of DWT coefficients from the bit stream. Tree-based techniques such as SPIHT or EZW accomplish this by reforming the set of significance maps from the bit stream, then using them to recreate each bit plane. During this process, bit plane data which was truncated during compression is replaced with zeros. For images compressed using JPEG 2000, the integer representation of each coefficient is decoded from the bit stream and the quantized DWT coefficient values are obtained through the dequantization process. Finally, the inverse DWT of the image or image tile is computed and resulting pixel values are projected back into the set $\mathcal{P}$ of allowable pixel values. If tiling was used, the full image is reassembled from the set of reconstructed tiles.

While these image compression techniques achieve lossy compression through different processes, they each introduce DWT coefficient quantization fingerprints into an image. For JPEG 2000, this is fairly obvious as the quantization and dequantization process causes the DWT coefficients in decompressed images to cluster around integer multiples of their respective subband's quantization step size. In SPIHT and related algorithms, a similar process occurs because bit stream truncation results in the loss of the bottom several bit planes. As a result, only the n most significant bits of each DWT coefficient are retained. This is equivalent to applying the

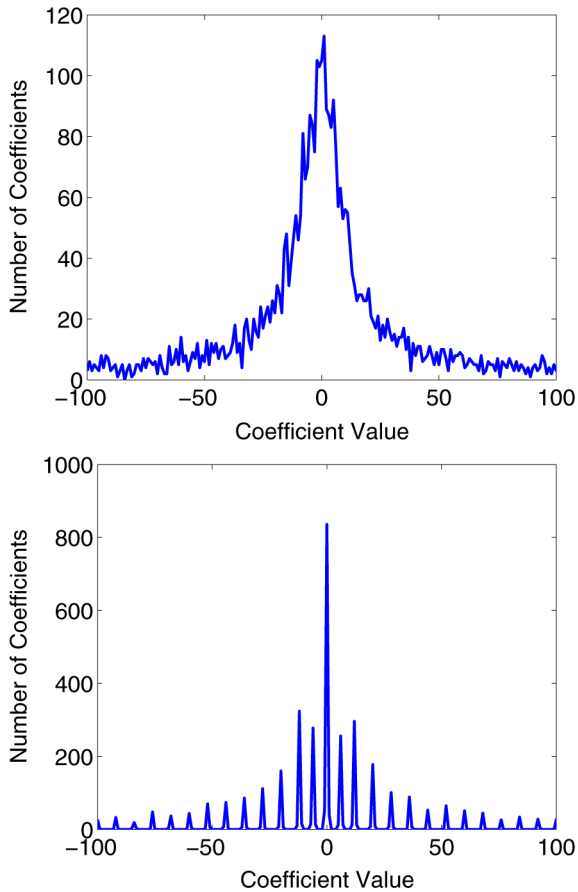

Fig. 4. Top: Histogram of wavelet coefficients from an uncompressed image. Bottom: Histogram of wavelet coefficients from the same image after SPIHT compression.

quantization rule in (2) where X is a DWT coefficient from an uncompressed image, Y is the corresponding DWT coefficient in its SPIHT compressed counterpart, and

$$q_k = \begin{cases} b_k, & \text{if } k \geq 1 \\ 0, & \text{if } k = 0 \\ b_{k+1}, & \text{if } k \leq -1. \end{cases} \tag{17}$$

These DWT coefficient quantization fingerprints can be observed when viewing the distribution of coefficient values within a particular DWT subband as seen in Fig. 4. Additionally, if the image was tiled during compression, tiling artifacts similar to JPEG blocking artifacts may occur in an image.

B. DWT Coefficient Quantization Fingerprint Removal

We model the distribution of coefficient values within a DWT subband of an uncompressed image using the Laplace distribution [23]

$$P(X = x) = \frac{\lambda}{2} e^{-\lambda |x|}. \tag{18}$$

Because the manner in which JPEG 2000 employs DWT coefficient quantization is identical to the way in which JPEG performs DCT coefficient quantization, the distribution of coefficients within a particular DWT subband in a previously JPEG 2000 compressed image is given by (10), where $Q_{i,j}$ is replaced by the quantization step size used for that DWT subband. As a result, the DWT coefficients in a previously JPEG 2000 compressed image can be anti-forensically modified using the method outlined in Section III-B. The remainder of this section will focus primarily on SPIHT and other tree-based compression schemes whose DWT coefficient quantization rules are given by (2) and (17). Using these equations along with (18), the distribution of coefficient values within a DWT subband in an image previously SPIHT or similarly compressed is given by

$$P(Y = q_k) = \begin{cases} \frac{1}{2}(e^{-\lambda b_k} - e^{-\lambda b_{k+1}}), & \text{if } k \geq 1 \\ 1 - \frac{1}{2}(e^{\lambda b_0} + e^{-\lambda b_1}), & \text{if } k = 0 \\ \frac{1}{2}(e^{\lambda b_{k+1}} - e^{\lambda b_k}), & \text{if } k \leq -1. \end{cases} \tag{19}$$

In order to anti-forensically modify an image previously compressed using a wavelet-based technique, we must first obtain the set of quantized DWT coefficients from the compressed image. To do this, we repeat the first several steps of compression including tiling the image if necessary and computing the DWT of the image or set of image tiles. Since the process of projecting the decompressed pixel values back into $\mathcal{P}$ during decompression perturbs the DWT coefficient values, the quantized coefficient values Y must be recovered from the perturbed coefficient values Y'. This can be done for previously JPEG 2000 compressed images by simply reapplying the quantization rule used during compression. For images compressed using SPIHT and related techniques, this is not appropriate since DWT coefficients are quantized to values on the edge of each quantization interval and the perturbations can move these values into a different quantization interval. Instead, we assume that the perturbations are sufficiently small and recover the quantized coefficient values according to the rule $Y = q_k$ if $(q_k + q_{k-1})/2 \leq Y' < (q_{k+1} + q_k)/2$.

Once the quantized DWT coefficient values have been recovered, we estimate the parameter λ for each DWT subband. This is done by fitting the nonzero entries of the histogram of each DWT subband's coefficient values to the function

$$h_k = c e^{-\hat{\lambda}|q_k|} \tag{20}$$

where $\hat{\lambda}$ is the estimated value of λ, h_k denotes the histogram value at q_k, and c is a scaling constant. By linearizing (20) by taking the logarithm of both sides of the equation, this fitting problem can be reformulated as the least squares minimization problem

$$\min_{\hat{\lambda}, c} \sum_k h_k (\log h_k - \log c + \hat{\lambda}|q_k|)^2 \tag{21}$$

where the model errors have been weighted by h_k, the number of observations of each quantized DWT coefficient value. To solve this minimization problem, we take the derivative with respect to $\hat{\lambda}$ and c of the function to be minimized in (21), set these derivatives to zero, then reformulate the resulting equations into the matrix

$$\begin{bmatrix} \sum_k h_k & \sum_k |q_k| h_k \\ \sum_k |q_k| h_k & \sum_k |q_k|^2 h_k \log h_k \end{bmatrix} \begin{bmatrix} \log c \\ -\hat{\lambda} \end{bmatrix} = \begin{bmatrix} \sum_k h_k \log h_k \\ \sum_k |q_k| h_k \log h_k \end{bmatrix}. \tag{22}$$

We then solve (22) for $\hat{\lambda}$ and c.

Though this estimate yields satisfactory results under ideal circumstances, in practice bit stream truncation effects often

lead to a mismatch between our model of the DWT coefficient distribution and the histogram of DWT coefficient values obtained from a previously compressed image. Because the point at which the bit stream is truncated rarely corresponds to the boundary between bit planes, a significant number of entries in the lowest bit plane are often set to zero. This results in an artificial decrease in the number of DWT coefficients taking the values q_1 and q_{-1}, and an artificial increase in the number of coefficients taking the value 0 over the number of coefficients predicted by our model. This, in turn, leads to an estimation bias which we compensate for using an iterative process to refine our estimate of λ.

We initialize our iterative procedure by setting $h_k^{(1)} = h_k$ for all k, $\hat{\lambda}^{(0)} = 0$, and the initial iteration index to $i = 1$. We then repeat the following steps until the termination criteria is met:

1) Estimate $\hat{\lambda}^{(i)}$ and $c^{(i)}$ by solving (22) using the current histogram iterate $\hat{h}^{(i)}$ in lieu of h.
2) Update the histogram estimate according to the equation

$$\hat{h}_k^{(i+1)} = \begin{cases} c^{(i)}, & \text{if } k = 0 \\ h_k + \frac{1}{2}\left(h_0 - c^{(i)}\right), & \text{if } k = \pm 1 \\ h_k, & \text{otherwise.} \end{cases} \tag{23}$$

3) Terminate if $(\hat{\lambda}^{(i)} - \hat{\lambda}^{(i-1)})/\hat{\lambda}^{(i)} < \tau$, where τ is a user defined threshold. Otherwise, set $i = i + 1$ and return to Step 1.

After this process is terminated, the final value of $\hat{\lambda}^{(i)}$ is retained as the parameter estimate $\hat{\lambda}$.

Before anti-forensic dither can be added to the quantized DWT coefficients, the mismatch between our model of the DWT coefficient distribution and the true DWT coefficient histogram must be corrected. Because bit stream truncation can occur anywhere within the least significant retained bit plane, we cannot accurately predict the number of components of that bit plane that will be set to zero. Accordingly, we cannot appropriately adjust our model to take partial bit plane truncation into account. Instead, we modify the DWT coefficient histogram to match our model by changing a number of DWT coefficient values from 0 to q_1 or q_{-1}. We calculate N_e, the number of zero valued DWT coefficients in excess of what our model predicts using the equation

$$N_e = h_0 - N_s\left(1 - \frac{1}{2}(e^{\hat{\lambda}b_0} + e^{-\hat{\lambda}b_1})\right) \tag{24}$$

where N_s is the total number of DWT coefficients in the current subband. We then randomly change the values of $N_e/2$ zero valued DWT coefficients to q_1 and $N_e/2$ zero valued coefficients to q_{-1}. After this modification, the DWT coefficient distribution should theoretically match our model.

Once a value of $\hat{\lambda}$ has been obtained for a DWT subband and the necessary histogram modifications have been performed, we generate the anti-forensic dither which is added to each DWT coefficient. As was the case with anti-forensic dither designed to modify JPEG compressed images, the use of the Laplace distribution to model the distribution of DWT coefficient values in an uncompressed image allows the anti-forensic dither distribution to be expressed using one of two equations. An analogous reduction in the complexity of generating the dither is realized as well, since once again the dither is drawn from only two distributions. The appropriate expression for the anti-forensic dither distribution depends upon the magnitude of the DWT coefficient to which it is added. When modifying nonzero valued DWT coefficients, the anti-forensic dither's distribution is given by

$$P(D = d \,|\, Y = q_k, k \neq 0) = \begin{cases} \frac{1}{\alpha_k} e^{-\mathrm{sgn}(q_k)\hat{\lambda}d}, & \text{if } (b_k - q_k) \leq d < (b_{k+1} - q_k) \\ 0, & \text{otherwise} \end{cases} \tag{25}$$

where $\alpha_k = (1/\hat{\lambda})(e^{-\mathrm{sgn}(q_k)\hat{\lambda}(b_k - q_k)} - e^{-\mathrm{sgn}(q_k)\hat{\lambda}(b_{k+1} - q_k)})$. When modifying zero valued DWT coefficients, the anti-forensic dither's distribution is

$$P(D = d \,|\, Y = 0) = \begin{cases} \frac{1}{\alpha_0} e^{-\hat{\lambda}|d|}, & \text{if } b_0 > d > b_1 \\ 0, & \text{otherwise} \end{cases} \tag{26}$$

where $\alpha_0 = (1/\hat{\lambda})(2 - e^{-\hat{\lambda}b_1} - e^{\hat{\lambda}b_0})$.

Assuming that we accurately estimate our model parameter so that $\hat{\lambda} = \lambda$ and that we accurately correct for truncation effects, the distribution of the anti-forensically modified coefficients in each DWT subband matches the model uncompressed coefficient distribution. Following the framework outlined in (6), we can demonstrate this by using the law of total probability as well as (19), (25), and (26) to write (27), shown at the bottom of the page.

Additionally, we can place an upper bound on the absolute distance between a DWT coefficient from an image compressed

$$\begin{aligned} P(Z = z) &= \sum_k P(Z = z \,|\, Y = q_k) P(Y = q_k) \\ &= \sum_{k \leq -1} \frac{1}{\alpha_k} e^{\lambda(z - q_k)} \frac{1}{2}(e^{\lambda b_{k+1}} - e^{\lambda b_k}) \mathbb{1}(b_k \leq z < q_{k+1}) \\ &\quad + \frac{1}{\alpha_0} e^{-\lambda|z|}\left(1 - \frac{1}{2}(e^{\lambda b_0} + e^{-\lambda b_1})\right) \mathbb{1}(b_0 \leq z < b_1) \\ &\quad + \sum_{k \geq 1} \frac{1}{\alpha_k} e^{-\lambda(z - q_k)} \frac{1}{2}(e^{-\lambda b_k} - e^{-\lambda b_{k+1}}) \mathbb{1}(b_k \leq z < q_{k+1}) \\ &= \frac{\lambda}{2} e^{-\lambda|z|} \end{aligned} \tag{27}$$

using a wavelet-based technique and its uncompressed counterpart. For images compressed using SPIHT or related techniques, the addition of anti-forensic dither will not move a DWT coefficient outside of its quantization interval, with the exception of zero-valued coefficients which remain in the interval $[b_{-1}, b_2)$. Since the corresponding uncompressed DWT coefficient must lie in the same interval, the upper bound on this distance becomes

$$|X - Z| \leq \begin{cases} b_{k+1} - b_k, & \text{if } k \neq 1 \\ b_2 - b_{-1}, & \text{if } k = 0 \end{cases} \tag{28}$$

given $b_k \leq X \leq b_{k+1}$. Because JPEG 2000 applies uniform quantization to the coefficient values within each DWT subband, we can use the upper bound given in (8) to write

$$|X - Z| \leq Q \tag{29}$$

where Q is the quantization step size used to quantize the coefficients that DWT subband.

V. Anti-Forensic Blocking Artifact Removal

As was discussed in Section II, if an image is divided into segments during compression, discontinuities are often present across segment boundaries in the decompressed image. These compression fingerprints, known as blocking artifacts, are commonly present in JPEG compressed images and can arise in JPEG 2000 compressed image if tiling is used. Even when blocking artifacts are not visually discernible, they can still be statistically detected [6]. Though the application of anti-forensic dither to an image removes transform coefficient quantization fingerprints, it does not remove blocking artifacts. If a previously compressed image is to be represented as never having undergone compression, these fingerprints must be removed.

While the removal of JPEG blocking artifacts is a well studied problem [24], [25], these techniques are designed to remove visible traces of blocking from low- to mid-quality images. To be successful, an anti-forensic deblocking technique must remove all visual and statistical traces of blocking artifacts without resulting in forensically detectable changes to an image's transform coefficient distributions or introducing new, forensically detectable fingerprints. Because existing deblocking algorithms are not designed to account for these criteria, they are poorly suited for anti-forensic purposes.

In order to remove statistical traces of blocking artifacts, we propose an anti-forensic deblocking technique that operates by first median filtering an image then adding low-power white Gaussian to each of its pixel vales. Letting $u_{i,j}$ and $v_{i,j}$ denote the value of a pixel at location (i, j) in an unmodified image and its deblocked counterpart, respectively, our anti-forensic deblocking operation can be expressed as

$$v_{i,j} = \text{med}_s(u_{i,j}) + n_{i,j} \tag{30}$$

where $n_{i,j}$ is a zero mean Gaussian random variable with variance σ^2. In this equation, med_s denotes a two-dimensional median filter with a square window of size s pixels, explicitly defined as $\text{med}_s(u_{i,j}) = \text{median}\{u_{l,m} | 0 \leq \lfloor (i-l)/2 \rfloor \leq s, 0 \leq \lfloor (j-m)/2 \rfloor \leq s\}$. We choose to use a median filter instead of a linear low-pass filter because its edge preserving nature tends to result in less visual distortion than simple linear filters. Both the window size of the median filter and the variance of the noise can be tuned according to the strength of the blocking artifacts present in the image. Heavily compressed images require the use of a larger median filter window size and greater noise variance to remove statistical traces of blocking artifacts than lightly compressed images. We compare the anti-forensic performance of this technique to those of existing deblocking algorithms in Section VI-C.

Fig. 5. Top: JPEG compressed image using a quality factor of 65. Bottom: Anti-forensically modified version of the same image.

VI. Experimental Results

In order to verify the efficacy of each of our proposed anti-forensic techniques, we have conducted a number of experiments in which we use our anti-forensic techniques to remove compression fingerprints from a set of images, then test each image for evidence of prior compression using several existing forensic techniques. In this section, we present the results of these experiments and analyze the performance of each proposed anti-forensic technique.

A. JPEG Anti-Forensics

To demonstrate that our anti-forensic DCT coefficient quantization fingerprint removal technique can be used on an image

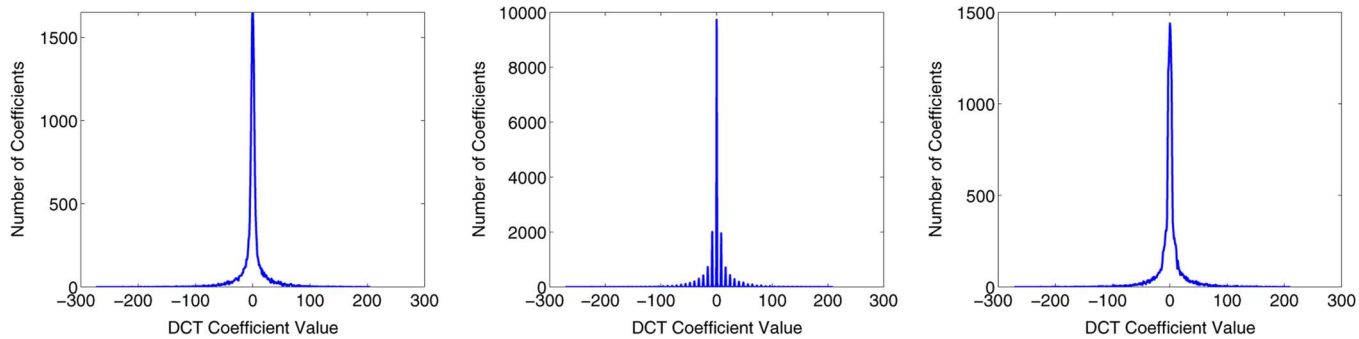

Fig. 6. Histogram of coefficient values from the (2,2) DCT subband taken from an uncompressed version of the image shown in Fig. 5 (left), the same image after JPEG compression (center), and an anti-forensically modified copy of the JPEG compressed image (right).

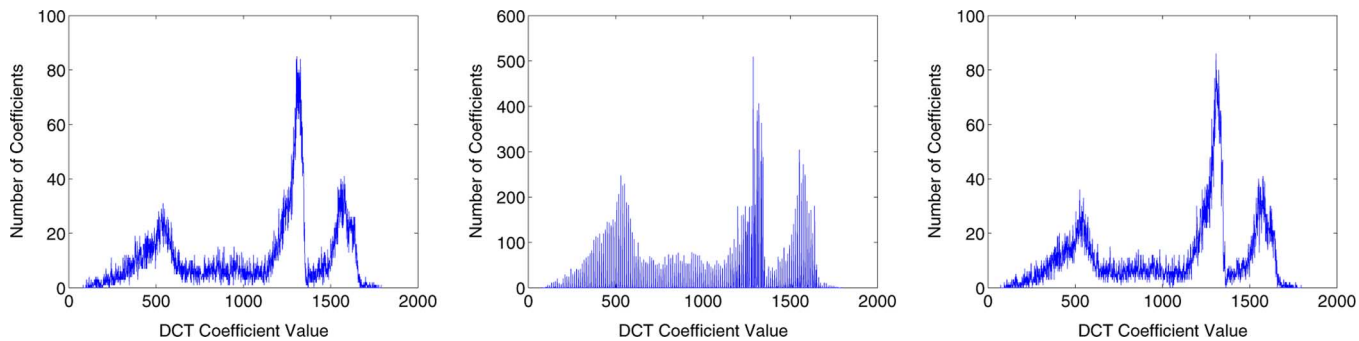

Fig. 7. Histogram of coefficient values from the dc DCT subband taken from an uncompressed version of the image shown in Fig. 5 (left), the same image after JPEG compression (center), and an anti-forensically modified copy of the JPEG compressed image (right).

without significantly impacting its visual quality, we show a typical image before and after anti-forensic modification in Fig. 5. In this figure, the image on top has undergone JPEG compression using a quality factor of 65 while the image on the right is the JPEG compressed image after anti-forensic dither has been added to its DCT coefficients. No noticeable difference between these images is apparent after visual inspection. This is reinforced by the fact that the PSNR between the two images is 41.63 dB. More importantly, the anti-forensically modified image contains no visual indicators of either previous compression or anti-forensic modification. Since a forensic examiner will not have access to either the unaltered or compressed version of an anti-forensically modified image, these cannot be compared against the anti-forensically modified image. Instead, what is necessary is that the anti-forensically modified image plausibly appear to have never been compressed.

Inspection of the DCT coefficient value distributions of the images shown in Fig. 5 yields similar results. Fig. 6 shows a histogram of coefficient values in the (2,2) DCT subband in an uncompressed version of these images along with the corresponding coefficient value histograms from the JPEG compressed and anti-forensically modified images. Fig. 7 shows the histogram of coefficient values in the dc DCT subband of the same images. While DCT coefficient quantization fingerprints are present in the histograms taken from the JPEG compressed image, these fingerprints are absent in the coefficient value histograms corresponding to the uncompressed and anti-forensically modified images. Again, we note that in reality a forensic examiner will only have access to the anti-forensically modified image and will be unable to make note of minor differences between the coefficient histograms of the uncompressed and anti-forensically modified image. The fact that the DCT coefficient value histograms from the anti-forensically modified image both fit our coefficient distribution model and contain no compression fingerprints suggests that our proposed anti-forensic technique is capable of producing images that can be passed off as never having undergone JPEG compression.

To verify that our anti-forensic technique is able to produce images that can fool existing forensic compression detection techniques, we conducted the following larger scale experiment. First, we converted each of the 1338 images in the Uncompressed Colour Image Database [26] to gray-scale, then we compressed each image using a quality factor of 90, 70, and 50. Next, we removed DCT coefficient quantization fingerprints from the JPEG compressed images by adding anti-forensic dither to the DCT coefficients of each image. Each of the anti-forensically modified images was then tested for DCT coefficient quantization fingerprints using the forensic technique developed by Fan and de Queiroz [6]. This technique detects previous applications of JPEG compression by using the DCT coefficient distributions to estimate the quantization step size used in each DCT subband during JPEG compression. If no evidence of quantization is present in any DCT subband, the image is classified as never-compressed. When we used this technique to search for evidence of JPEG compression in the anti-forensically modified images, it classified each anti-forensically modified image as never-compressed regardless of the quality factor used during compression. These results correspond to a 100% success rate for our anti-forensic DCT coefficient quantization fingerprint removal technique on this data set.

Fig. 8. Left: An image compressed using the SPIHT algorithm at a bit rate of 3 bits per pixel before the use of entropy coding. Right: The same image after anti-forensic dither has been applied to its wavelet coefficients.

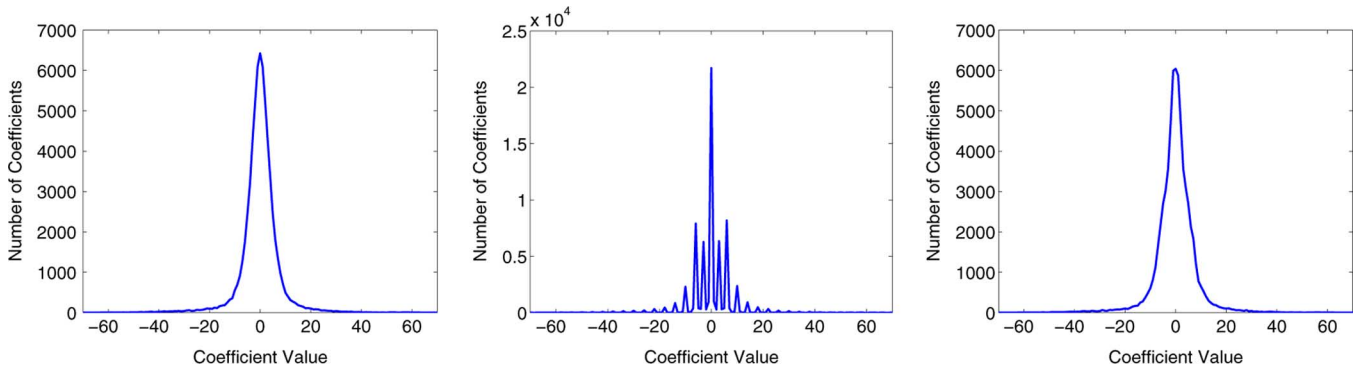

Fig. 9. Histogram of wavelet coefficients from the fourth level HH subband of a four level wavelet decomposition of the image shown in Fig. 8 (left), the same image after SPIHT compression (center), and the compressed image after anti-forensic dither has been applied (right).

B. Wavelet Anti-Forensics

We conducted a set of experiments similar to those in Section VI-A on SPIHT compressed images to demonstrate the effectiveness of our anti-forensic DWT coefficient compression fingerprint removal technique. Fig. 8 shows a version of the "Lena" image compressed at a bit rate of 3.0 bpp using the SPIHT algorithm both before and after anti-forensic dither has been added to its DWT coefficients. As was the case in our JPEG compression anti-forensics example, the two images contain no discernible differences and the anti-forensically modified image shows no signs of compression or anti-forensic modification. Furthermore, the PSNR between these two images is 46.64 dB. This result suggests that our anti-forensic DWT coefficient compression fingerprint removal technique will create images containing no visual indicators of compression or anti-forensic modification.

Fig. 9 shows the DWT coefficient histograms obtained from the fourth level HH subband of an uncompressed copy of the "Lena" image as well as from the SPIHT compressed and anti-forensically modified versions shown in Fig. 8. We note that the compression fingerprints observed in the DWT coefficient histogram from the SPIHT compressed image are absent from the DWT coefficient histogram corresponding to the anti-forensically modified image. This, along with the fact that the anti-forensically modified image's DWT coefficient histogram matches our coefficient distribution model, demonstrates that our anti-forensic DWT coefficient compression fingerprint removal technique is capable of modifying images so that they can be passed off as never having undergone wavelet-based compression.

In addition to the experimental results discussed above, we conducted a large-scale experiment to demonstrate that our anti-forensic technique is capable of misleading existing forensic wavelet-based compression detection algorithms. To do this, we again converted each of the 1338 images in the Uncompressed Colour Image Database [26] to gray-scale, then compressed them using the SPIHT algorithm at a bit rate of 2.0 bpp. We then removed image compression fingerprints from each image by adding anti-forensic dither to each image's DWT coefficients. Finally, we used the compression detection technique developed by Lin *et al.* [7] to test each image for evidence of prior wavelet-based compression. This detector was trained using the uncompressed and SPIHT compressed images, resulting in a classification rule that was able to correctly identify 99.8% of the SPIHT compressed images while only mis-

classifying 2.0% of the uncompressed images. When we used the trained wavelet-compression detection algorithm to classify the set of anti-forensically modified images, it was only able to correctly classify 1.2% of them as having undergone compression, resulting in a 98.2% success rate for our anti-forensic DWT compression fingerprint removal technique.

C. Anti-Forensic Deblocking

To evaluate our anti-forensic deblocking algorithm, we conducted an experiment in which we used it to remove blocking artifacts from several anti-forensically modified images, then compared its performance with those of the JPEG deblocking algorithms proposed by Liew and Yan [25], and Zhai *et al.* [25]. To perform this experiment, we first converted to gray-scale and JPEG compressed each of the 1338 images in the Uncompressed Colour Image Database using quality factors of 90, 70, 50, 30, and 10, then applied anti-forensic dither to the DCT coefficients of each of the compressed images. This created a testing database of 6690 anti-forensically modified gray-scale images. Next, we used our anti-forensic deblocking algorithm along with the deblocking algorithms proposed by Liew and Yan, and Zhai *et al.* to remove JPEG blocking artifacts from each image.

We tested each of the deblocked images for JPEG blocking fingerprints using the test designed by Fan and de Queiroz [6]. This method operates by collecting two pixel difference measurements throughout an image, one taken at the center of each block, which we refer to as R_1 and a second, which we refer to as R_2, taken across the boundary that occurs at the corners of each set of four adjacent blocks. Next, histograms of the R_1 and R_2 values obtained throughout the image, denoted h_1 and h_2, respectively, are tabulated. Finally, a test statistic K measuring the difference between the two histograms is computed according to the equation

$$K = \sum_r |h_1(r) - h_2(r)| \tag{31}$$

and K is compared to a threshold. If K is greater than the threshold, the image is classified as one which contains blocking artifacts.

We used the uncompressed and JPEG compressed images from our database to train this forensic blocking artifact detector and selected a decision threshold corresponding to a 99.1% probability of detecting blocking artifacts with a false detection rate of 0.0%. The trained detector was then used to test each of the deblocked images for blocking artifacts. Block artifact detection rates obtained from this experiment are shown in Table I. As we can see from this table, the deblocking methods of Liew and Yan, and Zhai *et al.* are poorly suited for removing statistical trace of blocking fingerprints from compressed images. By contrast, if the parameters s and σ^2 are properly chosen, our proposed algorithm is capable of removing statistical traces of blocking artifacts from images previously JPEG compressed at quality factors of 30 and above.

Additionally, we have discovered that existing deblocking techniques leave behind their own fingerprint. We have observed that under normal circumstances, $h_1(1) < h_1(0)$ and $h_2(1) < h_2(0)$ in an uncompressed image. This can be seen in Fig. 10 which shows the R_1 and R_2 histograms obtained from a typical image before and after JPEG quantization as well as after the JPEG compressed image was deblocked using our anti-forensic technique and those proposed by Zhai *et al.*, and Liew and Yan. By contrast, $h_1(1) > h_1(0)$ and $h_2(1) > h_2(0)$ in images deblocked using the Zhai *et al.*, and Liew and Yan techniques. This histogram feature can be used as a fingerprint indicating that an image has been deblocked using one of these algorithms. As Fig. 10 shows, this fingerprint is not present in images modified by our anti-forensic deblocking technique, indicating that it is much better suited for anti-forensic purposes.

TABLE I
BLOCKING ARTIFACT DETECTION RATES

Quality Factor	Proposed Method $s=3, \sigma^2=3$	Proposed Method $s=3, \sigma^2=2$	Proposed Method $s=2, \sigma^2=2$	Liew & Yan [25]	Zhai *et al.* [24]
90	0.0%	0.0%	0.2%	52.5%	98.3%
70	0.0%	0.1%	0.8%	76.3%	96.3%
50	0.2%	0.2%	1.6%	96.8%	96.1%
30	0.3%	10.5%	24.1%	99.2%	93.7%
10	49.0%	79.0%	95.9%	99.0%	70.8%

Though our anti-forensic dither and deblocking techniques can successfully remove statistical traces of compression artifacts from heavily compressed images, they cannot compensate for significant visual distortion caused by the initial application of compression. For heavily compressed images, they can serve to significantly increase the distortion present in an image. Fig. 11 shows a typical image after JPEG compression using several quality factors followed by the anti-forensic removal of both its DCT coefficient quantization fingerprints and its blocking fingerprints. While the images compressed using quality factors of 70 and 90 appear to be unaltered, the images compressed with a quality factor of 30 and below contain noticeable distortions. Accordingly, as an image is more heavily compressed, it is more difficult to convincingly disguise both visual and statistical traces of its compression history.

VII. UNDETECTABLE IMAGE TAMPERING USING ANTI-FORENSICS

In many scenarios, an image forger is not concerned with representing a previously compressed image as one which has never undergone compression. More likely, the image forger wishes to alter an image, then remove all evidence that the image has been manipulated. In such a scenario, the image forger must pay particular attention to an image's compression history. Because most digital cameras store images as JPEGs by default, many digital images are imprinted with compression fingerprints at the time of their capture. If an image contains evidence that it has been compressed multiple times, this suggests that the image has been decompressed for editing, then saved again in a compressed format. If the forger attempts to avoid the fingerprints left by multiple applications of compression by saving an image in an uncompressed format after editing, the fingerprints left by the initial application of compression will reveal evidence of image manipulation. Furthermore, spatial inconsistencies in an image's compression fingerprints are often used as forensic evidence of cut-and-paste forgery, in which a composite image is formed by cutting an object from one image,

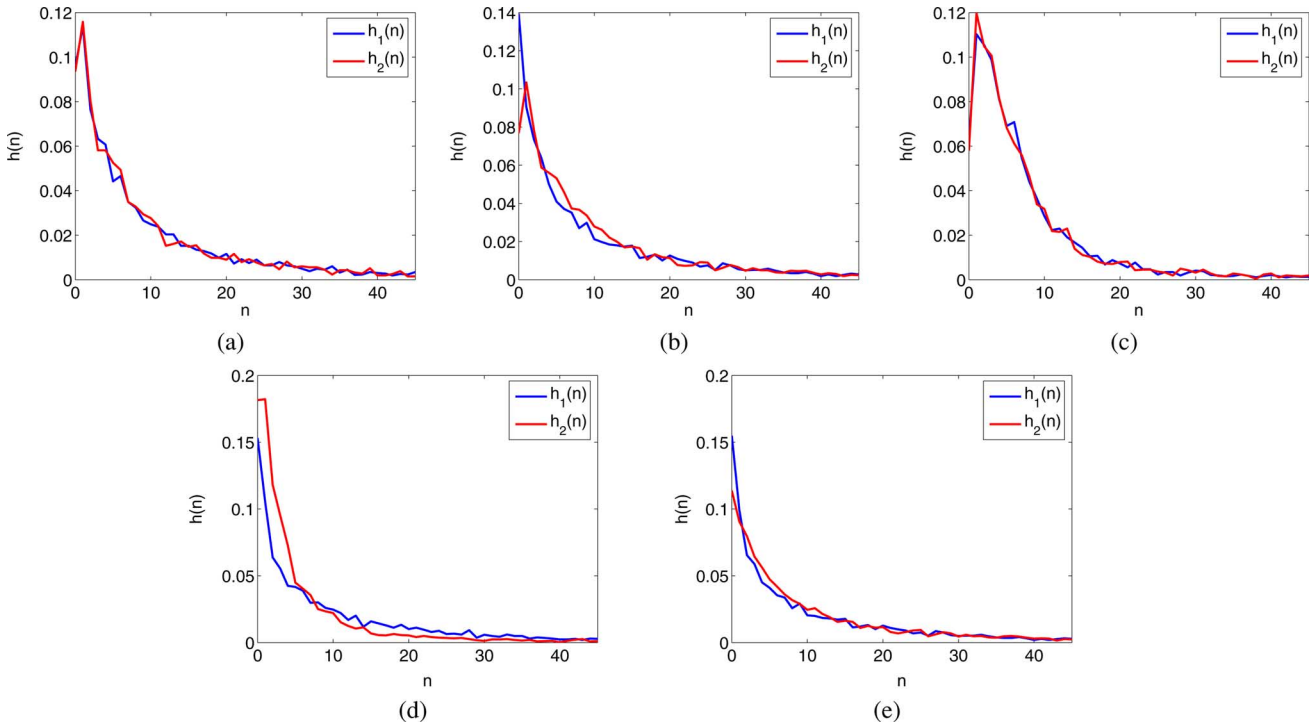

Fig. 10. Histograms of R_1 and R_2 blocking artifact detection statistics obtained from (a) an uncompressed image, (b) the same image after JPEG compression using a quality factor of 70, as well as the JPEG compressed version after it has been deblocked using (c) our anti-forensic deblocking algorithm, (d) the deblocking algorithm proposed by Liew and Yan, and (e) the deblocking algorithm proposed by Zhai *et. al*.

then pasting it into another. If used properly, the anti-forensic techniques outlined in this paper can either remove or prevent the occurrence of each of these image tampering fingerprints.

Recompression of a JPEG image, commonly referred to as double JPEG compression, introduces a unique fingerprint in an image's DCT coefficient distributions. During the initial application of JPEG compression, DCT coefficient quantization causes an image's DCT coefficients to cluster around integer multiples of a particular DCT subband's quantization step size. When the image is compressed a second time using a different quantization table, some DCT subbands will be quantized using a different quantization step size. This mismatch in quantization step sizes will cause an unequal number of DCT coefficient clusters to fall within each new quantization interval. As a result, the DCT coefficient distributions of a double JPEG compression will appear to be modulated by a periodic signal. A number of forensic techniques use this signal to identify double JPEG compression [9], [10].

To prevent double JPEG compression fingerprints from occurring in a doubly compressed image, an image forger can add anti-forensic dither to a singly compressed image's DCT coefficients before it is recompressed. By doing this, the image's DCT coefficients will be distributed as if they came from an uncompressed image rather than being clustered around integer multiples of the first quantization step size. When the image is recompressed, quantization interval mismatch effects will not occur, allowing the double JPEG compressed image's DCT coefficients to be distributed as if they came from an image compressed only once. Since the image will remain JPEG compressed in its final state, it does not need to be anti-forensically deblocked.

An example demonstrating that anti-forensic dither can be used to prevent double JPEG compression fingerprints is shown in Fig. 12. In this example, we show coefficient histograms from the (3,3) DCT subband of an image compressed once using a quality factor of 85, the same image after it has been double compressed using a quality factor 75 followed by 85, as well as the image compressed first with a quality factor 75, then anti-forensically modified and recompressed using a quality factor of 85. While double JPEG compression fingerprints can be observed in the coefficient histogram of the doubly JPEG compressed image that did not have anti-forensic dither added to its DCT coefficients, these fingerprints are absent from the coefficient histogram of the image that underwent anti-forensic modification. Additionally, the coefficient histogram of the anti-forensically modified double JPEG compressed image does not differ greatly from the coefficient histogram of the singly compressed image. This verifies that under forensic inspection, the anti-forensically modified image would appear to have only been compressed once.

If two JPEG compressed images are used to create a cut-and-paste forgery, the composite image will contain double JPEG compression fingerprints that differ spatially. These locally varying fingerprints can be used to both detect forged images and to identify falsified image regions [11]. Alternately, if blocking artifacts in the pasted region do not align with those throughout the rest of the image, the resulting mismatch in the blocking grid can be used to detect cut-and-paste forgeries [12]. Both of these fingerprints can be avoided if the two images used to create the forgery have anti-forensic dither added to their DCT coefficients and are anti-forensically deblocked before the composite image is created. Doing this will render

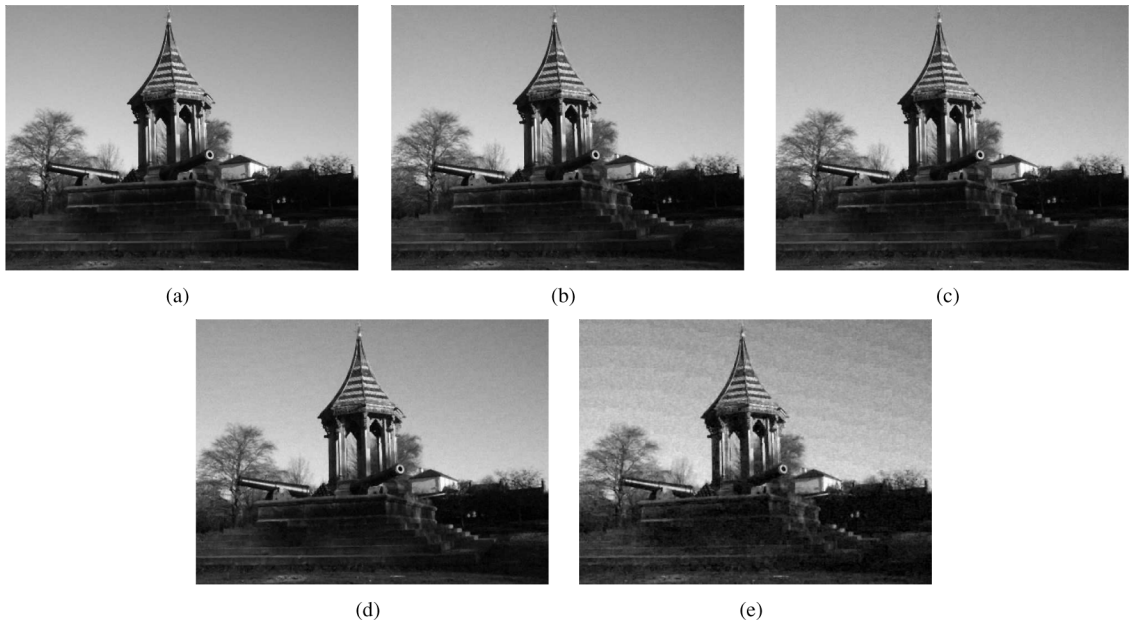

Fig. 11. Results of the proposed anti-forensic deblocking algorithm applied to a typical image after it has been JPEG compressed using a quality factor of (a) 90, (b) 70, (c) 50, (d) 30, and (e) 10 followed by the addition of anti-forensic dither to its DCT coefficients.

compression-history-based forensic techniques unable to detect cut-and-paste image forgeries.

In other situations, an image forger may wish to falsify the origin of an image. Since most digital cameras and image editing software use proprietary JPEG quantization tables when storing images, the camera model used to capture an image can be determined by identifying the image's quantization table in a list of camera and quantization table pairings [8]. This means that information about an image's origin is intrinsically embedded in an image via its compression history. Software designed to perform quantization table and camera matching known as *JPEGsnoop* is readily available online [27]. As a result, an image forger cannot mislead forensic investigators by simply changing an image's metadata tags. While other forensic signatures such as a camera's sensor noise [1] and color filter array interpolation parameters [3] can be used as a means of camera identification, these techniques can be defeated by falsifying the sensor noise pattern [14] and reapplying the color filter array then reinterpolating the image [15] respectively.

An image's origin cannot be forged by simply recompressing it using the quantization table of another camera. Doing this will result in double JPEG compression artifacts that can alert forensic investigators to the fact that the image has been tampered with. Instead, we are able to undetectably falsify the compression history aspects of an image's origin by first removing traces of prior JPEG compression through the use of anti-forensic dither, then compressing the image with the quantization table of another camera.

To verify that our anti-forensic technique is suitable for image origin forgery purposes, we conducted an experiment in which we falsified the compression signatures of images taken by several cameras, then attempted to link each image with its origin using existing forensic techniques. For this experiment, we compiled a database consisting of 100 images from each of the following cameras: a Canon Powershot G7 (Cam 1), Sony Cybershot DSC-W80 (Cam 2), Sony Cybershot DSC-V1 (Cam 3), Fuji Finepix E550 (Cam 4), and an Olympus Camedia C5060 (Cam 5). We removed evidence of prior JPEG compression from each image by adding anti-forensic dither to its DCT coefficients, then recompressed it with the quantization tables used by each of the other cameras in the database. After this was done, we used the procedure developed by Fan and de Quieroz to obtain an estimate $\hat{Q}_{i,j}$ of the quantization table used to compress each image [6]. We matched each image with a camera by selecting the camera whose quantization table $Q_{i,j}^{(k)}$ maximized the similarity measure

$$s_k = \sum_i \sum_j \mathbb{1}\left(\hat{Q}_{i,j}, Q_{i,j}^{(k)}\right). \tag{32}$$

Table II shows the results of our image origin forgery experiment. With the exception of when we attempted to represent the images captured by the Sony Cybershot DSC-V1 originating from the Sony Cybershot DSC-W80, we were able to falsify the origin of the images captured by each camera with a 100% success rate. In the case of the Sony Cybershot DSC-V1, one image was unintentionally linked to a different camera than the one we intended.

VIII. Conclusion

In this paper, we have proposed a set of anti-forensic operations capable of removing compression fingerprints from digital images. To do this, we developed a generalized framework for the removal of quantization fingerprints from an image's transform coefficients. According to this framework, quantization fingerprints can be removed from an image's transform coefficients by first estimating the distribution of the image's transform coefficients before compression, then adding anti-forensic dither to the compressed image's transform

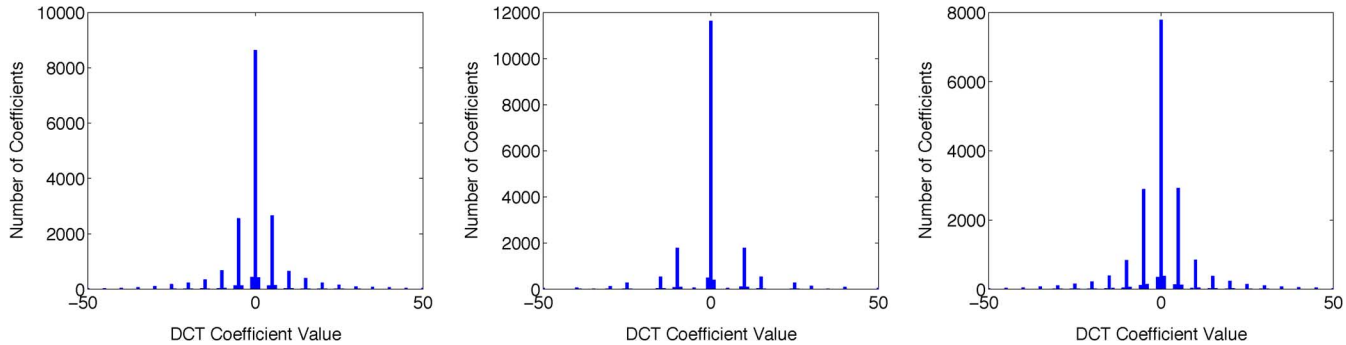

Fig. 12. Histogram of (3,3) DCT coefficients from an image JPEG compressed once using a quality factor of 85 (left), the image after being double JPEG compressed using a quality factor of 75 followed by 85 (center), and the image after being JPEG compressed using a quality factor of 75, followed by the application of anti-forensic dither, then recompressed using a quality factor of 85 (right).

TABLE II
CAMERA ORIGIN FORGERY CLASSIFICATION RESULTS

Falsified Origin	True Image Origin				
	Cam 1	Cam 2	Cam 3	Cam 4	Cam 5
Cam 1	-	100.0%	100.0%	100.0%	100.0%
Cam 2	100.0%	-	99.0%	100.0%	100.0%
Cam 3	100.0%	100.0%	-	100.0%	100.0%
Cam 4	100.0%	100.0%	100.0%	-	100.0%
Cam 5	100.0%	100.0%	100.0%	100.0%	-

coefficients so that their anti-forensically modified distribution matches the estimate of their distribution before compression. We used this framework to design specific anti-forensic techniques to remove DCT coefficient quantization artifacts from JPEG compressed images and DWT coefficient compression artifacts from images compressed using wavelet-based coders. Additionally, we have proposed an anti-forensic technique capable of removing statistical traces of blocking artifacts from images that undergo blockwise segmentation during compression.

To demonstrate the performance of our algorithms, we have conducted a number of experiments on JPEG and SPIHT compressed images in which we show that by adding anti-forensic dither to an image's transform coefficients, we can render that image's transform coefficient compression fingerprints forensically undetectable without significantly degrading the image's visual quality. We have conducted an experiment showing that our anti-forensic deblocking technique can remove statistical traces of blocking artifacts from images while several existing deblocking techniques cannot. Additionally, we have shown that our proposed anti-forensic techniques can be used to make certain types of image tampering such as double JPEG compression, cut-and-paste image forgery, and image origin falsification undetectable to compression-history-based forensic techniques.

REFERENCES

[1] M. Chen, J. Fridrich, M. Goljan, and J. Lukáš, "Determining image origin and integrity using sensor noise," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 1, pp. 74–90, Mar. 2008.

[2] J. Lukáš and J. Fridrich, "Estimation of primary quantization matrix in double compressed JPEG images," in *Proc. Digital Forensic Research Workshop*, Aug. 2003, pp. 5–8.

[3] A. Swaminathan, M. Wu, and K. J. R. Liu, "Digital image forensics via intrinsic fingerprints," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 1, pp. 101–117, Mar. 2008.

[4] I. Avcibas, S. Bayram, N. Memon, M. Ramkumar, and B. Sankur, "A classifier design for detecting image manipulations," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2004, vol. 4, pp. 2645–2648.

[5] M. C. Stamm and K. J. R. Liu, "Forensic detection of image manipulation using statistical intrinsic fingerprints," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 3, pp. 492–506, Sep. 2010.

[6] Z. Fan and R. de Queiroz, "Identification of bitmap compression history: JPEG detection and quantizer estimation," *IEEE Trans. Image Process.*, vol. 12, no. 2, pp. 230–235, Feb. 2003.

[7] W. S. Lin, S. K. Tjoa, H. V. Zhao, and K. J. R. Liu, "Digital image source coder forensics via intrinsic fingerprints," *IEEE Trans. Inf. Forensics Security*, vol. 4, no. 3, pp. 460–475, Sep. 2009.

[8] H. Farid, Digital Image Ballistics From JPEG Quantization Dept. of Computer Science, Dartmouth College, Tech. Rep. TR2006-583, 2006.

[9] A. C. Popescu and H. Farid, "Statistical tools for digital forensics," in *Proc. 6th Int. Workshop Information Hiding*, Toronto, Canada, 2004.

[10] T. Pevny and J. Fridrich, "Detection of double-compression in JPEG images for applications in steganography," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 2, pp. 247–258, Jun. 2008.

[11] J. He, Z. Lin, L. Wang, and X. Tang, "Detecting doctored JPEG images via DCT coefficient analysis," in *Proc. Eur. Conf. Computer Vision*, May 2006, vol. 3593, pp. 423–435.

[12] S. Ye, Q. N. Sun, and E.-C. Chang, "Detecting digital image forgeries by measuring inconsistencies of blocking artifact," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2007, pp. 12–15.

[13] M. Kirchner and R. Bohme, "Hiding traces of resampling in digital images," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 4, pp. 582–592, Dec. 2008.

[14] T. Gloe, M. Kirchner, A. Winkler, and R. Böhme, "Can we trust digital image forensics?," in *Proc. 15th Int. Conf. Multimedia*, 2007, pp. 78–86.

[15] M. Kirchner and R. Böhme, "Synthesis of color filter array pattern in digital images," in *Proc. SPIE-IS&T Electronic Imaging: Media Forensics and Security*, San Jose, CA, Feb. 2009, vol. 7254.

[16] M. C. Stamm, S. K. Tjoa, W. S. Lin, and K. J. R. Liu, "Anti-forensics of JPEG compression," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Mar. 2010, pp. 1694–1697.

[17] M. C. Stamm and K. J. R. Liu, "Wavelet-based image compression anti-forensics," in *Proc. IEEE Int. Conf. Image Process.*, Sept. 2010, pp. 1737–1740.

[18] M. C. Stamm, S. K. Tjoa, W. S. Lin, and K. J. R. Liu, "Undetectable image tampering through JPEG compression anti-forensics," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 2109–2112.

[19] E. Y. Lam and J. W. Goodman, "A mathematical analysis of the DCT coefficient distributions for images," *IEEE Trans. Image Process.*, vol. 9, no. 10, pp. 1661–1666, Oct. 2000.

[20] J. R. Price and M. Rabbani, "Biased reconstruction for JPEG decoding," *IEEE Signal Process. Lett.*, vol. 6, no. 12, pp. 297–299, Dec. 1999.

[21] A. Said and W. A. Pearlman, "A new fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 3, pp. 243–250, Jun. 1996.

[22] A. Skodras, C. Christopoulos, and T. Ebrahimi, "The JPEG 2000 still image compression standard," *IEEE Signal Process. Mag.*, vol. 18, no. 5, pp. 36–58, Sep. 2001.
[23] J. Li and R. M. Gray, "Text and picture segmentation by the distribution analysis of wavelet coeffcients," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 1998, pp. 790–794.
[24] A. W.-C. Liew and H. Yan, "Blocking artifacts suppression in block-coded images using overcomplete wavelet representation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 4, pp. 450–461, Apr. 2004.
[25] G. Zhai, W. Zhang, X. Yang, W. Lin, and Y. Xu, "Efficient image deblocking based on postfiltering in shifted windows," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 1, pp. 122–126, Jan. 2008.
[26] G. Schaefer and M. Stich, "UCID: An uncompressed color image database," in *Proc. SPIE: Storage and Retrieval Methods and Applications for Multimedia*, 2004, vol. 5307, pp. 472–480.
[27] C. Haas, JPEGsnoop—JPEG File Decoding Utility [Online]. Available: http://www.impulseadventure.com/photo/jpeg-snoop.html

Matthew C. Stamm (S'08) received the B.S. degree in electrical engineering in 2004 from the University of Maryland, College Park. He is currently working toward the Ph.D. degree at the Department of Electrical and Computer Engineering, University of Maryland, College Park.

From 2004 to 2006, he was a radar systems engineer at the Johns Hopkins University Applied Physics Laboratory. His current research interests include digital multimedia forensics and anti-forensics as well as music information retrieval.

Mr. Stamm received a Distinguished Teaching Assistant Award in 2006, a Future Faculty Fellowship in 2010, and an Ann G. Wylie Dissertation Fellowship in 2011 from the University of Maryland.

K. J. Ray Liu (F'03) is a Distinguished Scholar-Teacher of University of Maryland, College Park, in 2007, where he is Christine Kim Eminent Professor in Information Technology. He serves as Associate Chair of Graduate Studies and Research of the Electrical and Computer Engineering Department and leads the Maryland Signals and Information Group conducting research encompassing broad aspects of wireless communications and networking, information forensics and security, multimedia signal processing, and biomedical engineering.

Dr. Liu is the recipient of numerous honors and awards including IEEE Signal Processing Society Technical Achievement Award and Distinguished Lecturer. He also received various teaching and research recognitions from University of Maryland including university-level Invention of the Year Award; and Poole and Kent Senior Faculty Teaching Award and Outstanding Faculty Research Award, both from A. James Clark School of Engineering. An ISI Highly Cited Author in Computer Science, he a fellow of AAAS. He is President-Elect and was Vice President–Publications of IEEE Signal Processing Society. He was the Editor-in-Chief of *IEEE Signal Processing Magazine* and the founding Editor-in-Chief of *EURASIP Journal on Advances in Signal Processing*. His recent books include *Cognitive Radio Networking and Security: A Game Theoretical View* (Cambridge University Press, 2010); *Behavior Dynamics in Media-Sharing Social Networks* [Cambridge University Press (to appear)]; *Handbook on Array Processing and Sensor Networks* (IEEE-Wiley, 2009); *Cooperative Communications and Networking* (Cambridge University Press, 2008); *Resource Allocation for Wireless Networks: Basics, Techniques, and Applications* (Cambridge University Press, 2008); *Ultra-Wideband Communication Systems: The Multiband OFDM Approach* (IEEE-Wiley, 2007); *Network-Aware Security for Group Communications* (Springer, 2007); *Multimedia Fingerprinting Forensics for Traitor Tracing* (Hindawi, 2005).