

# UNDETECTABLE IMAGE TAMPERING THROUGH JPEG COMPRESSION ANTI-FORENSICS

Matthew C. Stamm, Steven K. Tjoa, W. Sabrina Lin, and K. J. Ray Liu

Dept. of Electrical and Computer Engineering, University of Maryland, College Park

## ABSTRACT

Recently, a number of digital image forensic techniques have been developed which are capable of identifying an image's origin, tracing its processing history, and detecting image forgeries. Though these techniques are capable of identifying standard image manipulations, they do not address the possibility that anti-forensic operations may be designed and used to hide evidence of image tampering. In this paper, we propose an anti-forensic operation capable of removing blocking artifacts from a previously JPEG compressed image. Furthermore, we show that by using this operation along with another anti-forensic operation which we recently proposed, we are able to fool forensic methods designed to detect evidence of JPEG compression in decoded images, determine an image's origin, detect double JPEG compression, and identify cut-and-paste image forgeries.

*Index Terms*— Anti-Forensics, Digital Forensics, JPEG Compression

## 1. INTRODUCTION

The widespread availability of software capable of creating visually convincing digital image forgeries has resulted in an environment where the authenticity of digital images can not be trusted. To combat this, a wide variety of digital image forensic techniques have been developed to identify an image's origin, trace its processing history, and detect image forgeries without relying on extrinsically embedded information such as metadata tags or watermarks.

Many of these digital forensic techniques rely on detecting artifacts left in an image by JPEG compression. Methods designed to detect previous instances of JPEG compression in images saved using uncompressed or losslessly compressed file formats have been developed [1], [2], as well as an algorithm capable estimating the quantization table employed during compression [1]. Because most digital cameras make use of proprietary quantization tables, an image's compression history can be used to help identify the camera used to capture it [3]. A second application of JPEG compression to an image previously JPEG compressed can be detected [4], [5]. Furthermore, techniques have been proposed to identify cut-and-paste image forgeries by detecting spatially localized discrepancies in an image's JPEG compression signature [6], [7].

Though these forensic techniques are quite adept at detecting standard image manipulations, they do not account for the possibility that *anti-forensic* operations designed to hide traces of image manipulation may be applied to an image. Recent work has shown that such operations can be constructed to successfully fool existing image forensic techniques [8]. In light of this, it is possible that image manipulators may be creating undetectable image forgeries by using secretly developed anti-forensic countermeasures. If this

situation is to be prevented, it is necessary that researchers develop and study anti-forensic operations so that vulnerabilities in existing forensic techniques may be known. Furthermore, by studying anti-forensic operations, researchers may be able to develop techniques capable of detecting anti-forensic manipulation.

In a recent paper, we proposed an anti-forensic technique capable of removing forensically significant JPEG compression artifacts from an image's DCT coefficient histograms [9]. In this paper, we introduce a simple technique designed to render JPEG blocking artifacts both visually and statistically undetectable without resulting in forensically detectable changes to an image's DCT coefficient histograms. Furthermore, we show how both of these techniques can be used to fool forensic algorithms designed to detect evidence of prior applications of JPEG compression within uncompressed images, determine an image's origin, detect multiple applications of JPEG compression, and identify cut and paste type image forgeries.

## 2. BACKGROUND

When an image is subjected to JPEG compression, it is first segmented into  $8 \times 8$  pixel blocks. The DCT of each block is computed and the resulting set of DCT coefficients are quantized by dividing each coefficient by its corresponding entry in a quantization table, then rounding the result to the nearest integer. Finally, the set of quantized coefficients are read into a single bitstream and losslessly encoded. Decompression begins by decoding the bitstream of quantized DCT coefficients and reforming into a set of  $8 \times 8$  pixel blocks. Each DCT coefficient is then dequantized by multiplying it by its corresponding entry in the quantization table. Finally, the inverse DCT of each block is performed and the resulting pixel values are projected into the set of allowable pixel values.

As a result of this process, two forensically significant artifacts are left in an image by JPEG compression; DCT coefficient quantization artifacts and blocking artifacts. DCT coefficient quantization artifacts correspond to the clustering of DCT coefficients around integer multiples of a particular quantization table entry that occur due to the coupling of the quantization and dequantization operations. This can be clearly seen when examining a histogram of an image's DCT coefficients as shown in Fig. 1. Blocking artifacts are the discontinuities which occur across  $8 \times 8$  pixel block boundaries because of JPEG's lossy nature. Both of these artifacts are used by several image forensic algorithms to trace an image's compression history, identify the device used to create the image, and identify composite image forgeries.

In [9], we proposed an anti-forensic technique capable of removing DCT coefficient artifacts from a previously compressed image. Our technique operates by adding noise, which we shall hereafter refer to as *anti-forensic dither*, to each DCT coefficient so that the distribution of anti-forensically modified DCT coefficients approximates the distribution of DCT coefficients before compression. The anti-forensic dither distribution is conditionally dependent upon both

the value of the DCT coefficient to which it is added as well as an estimate of the DCT coefficient distribution before compression. This estimate is obtained by using the Laplace distribution to parametrically model the distribution of DCT coefficients before compression, then using the set of quantized DCT coefficients to obtain a maximum likelihood estimate of the Laplace distribution's parameter. The bottom plot of Fig. 1 shows the histogram of JPEG compressed DCT coefficients after anti-forensic dither has been added to them.

### 3. ANTI-FORENSIC DEBLOCKING OPERATION

If a previously JPEG compressed image is to be passed off as never having undergone compression, JPEG blocking artifacts must be removed from the image after anti-forensic dither has been applied to its DCT coefficients. Though a number of deblocking algorithms have been proposed since the introduction of the JPEG compression standard, the majority of these are ill suited for anti-forensic purposes. In order for an anti-forensic deblocking operation to be successful, it must remove all visual and statistical traces of block artifacts without resulting in forensically detectable changes to an image's DCT coefficient histograms. By contrast, existing deblocking algorithms are designed to only remove visible traces of blocking artifacts, particularly in heavily compressed images, and do not give consideration to the forensic detectability of compression artifacts in their output images.

Experimentally, we have found that by lightly smoothing an image followed by adding low-power white Gaussian noise, we are able to remove statistical traces of JPEG blocking artifacts without causing the images DCT coefficient distribution to deviate from the Laplace distribution. In light of this, we propose the following deblocking algorithm which is suitable for anti-forensic purposes. Let  $x_{i,j}$  denote the pixel at location  $(i, j)$  in the image to be deblocked. We obtain each pixel value  $y_{i,j}$  in the anti-forensically deblocked image according to the equation

$$y_{i,j} = \text{med}_s(x_{i,j}) + n_{i,j} \quad (1)$$

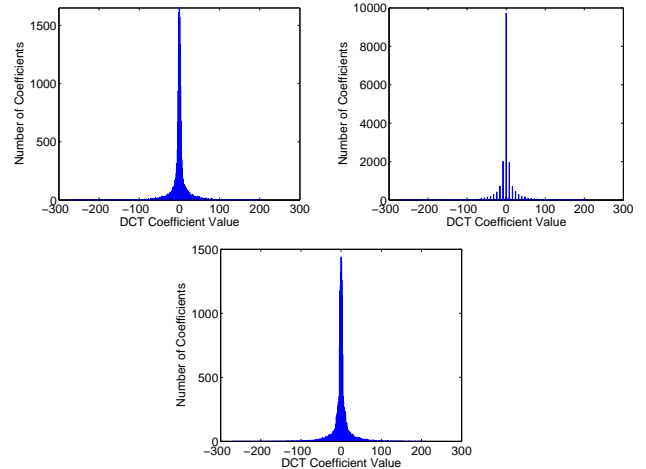
where  $\text{med}_s(x_{i,j}) = \text{median}\{x_{l,m} | 0 \leq \lfloor \frac{(s-l)}{2} \rfloor \leq s, 0 \leq \lfloor \frac{(s-m)}{2} \rfloor \leq s\}$  and  $n_{i,j}$  is a zero mean Gaussian random variable with variance  $\sigma^2$ . The parameters  $s$  and  $\sigma^2$  are user defined and can be chosen in accordance with the quality factor of the JPEG image to be deblocked. We use a median filter with a square window of size  $s$  to perform smoothing because its edge preserving nature tends to result in less visual distortion than simple linear filters.

To demonstrate the effectiveness of this anti-forensic deblocking operation as well as to illustrate its advantages over several existing deblocking algorithms, we have tested its ability to deceive the forensic JPEG blocking artifact detector proposed in [1] along with the deblocking algorithms recently proposed in [10] and [11]. This detector operates by obtaining a measure  $Z'$  of the pixel differences within a block along with a measure  $Z''$  of the pixel differences across block boundaries for each  $8 \times 8$  pixel block within an image. A measure of the blocking artifact strength is obtained by calculating the difference between the histograms of  $Z'$  and  $Z''$  values, denoted by  $H_I$  and  $H_{II}$  respectively, using the equation

$$K = \sum_n |H_I(Z' = n) - H_{II}(Z'' = n)|. \quad (2)$$

Values of  $K$  lying above a fixed detection threshold indicate the presence of blocking artifacts.

We created a training database by converting each of the 244 images in the Uncompressed Colour Image Database [12] from color to



**Fig. 1.** Histogram of DCT coefficients from an image before compression (Top Left), after JPEG compression (Top Right), and after the addition anti-forensic dither to the coefficients of the JPEG compressed image (Bottom).

grayscale, then JPEG compressing each image at quality factors of 90, 70, 50, 30, and 10. We then used this database to train the JPEG blocking artifact detector, selecting a decision threshold corresponding to a 95.9% probability of detecting blocking artifacts with a false detection rate of 0.0%. A testing database was created by applying our anti-forensic DCT artifact removal algorithm to each compressed image, then using the proposed deblocking operation along with the algorithms proposed in [10] and [11] to remove JPEG blocking artifacts. Each image in the testing database was then tested for JPEG blocking artifacts using the trained detector.

Table 1 shows JPEG blocking artifact detection results obtained from our tests. These results clearly demonstrate that when the parameters  $s$  and  $\sigma^2$  are chosen properly, our proposed algorithm is capable of removing statistical traces of blocking artifacts from images previously JPEG compressed at quality factors of 30 and above. Furthermore, these results indicate that while the algorithms presented in [10] and [11] are able to remove visual traces of blocking artifacts, they do not entirely remove all statistical traces and are not appropriate for anti-forensic purposes.

We should note that while our proposed operation is capable of removing statistical traces of blocking from images previously compressed at low quality factors, significant visual distortion introduced by compression will not be removed. Fig. 2 shows a typical image after compression using several different quality factors followed by the addition of anti-forensic dither and anti-forensic deblocking. The images previously compressed using quality factors of 30 and 10 lower exhibit noticeable visual distortions, suggesting they can not be convincingly passed off as never-compressed. By contrast, the images previously compressed using quality factors of 70 and 90 contain no visual indicators that the image was previously compressed or otherwise manipulated. In general, as an image is subjected to greater amounts of compression, it becomes more difficult to convincingly disguise its compression history.

### 4. IMAGE TAMPERING THROUGH ANTI-FORENSICS

In this section, we show how anti-forensic dither and our proposed anti-forensic deblocking operation can be used to deceive several existing image forensic algorithms that rely on detecting JPEG compression artifacts.



**Fig. 2.** Results of the proposed anti-forensic deblocking algorithm applied to a typical image after it has been JPEG compressed using a quality factor of 90 (Far Left), 70 (Center Left), 30 (Center Right), and 10 (Far Right) followed by the addition of anti-forensic dither to its DCT coefficients.

Quality Factor	Proposed Method			Liew & Yan [10]	Zhai <i>et al.</i> [11]
	$s = 3, \sigma^2 = 3$	$s = 3, \sigma^2 = 2$	$s = 2, \sigma^2 = 2$		
90	0.0%	0.0%	0.0%	70.1%	99.6%
70	0.0%	0.0%	14.8%	99.2%	99.6%
50	0.0%	0.9%	62.7%	98.8%	99.6%
30	3.3%	23.0%	93.4%	99.6%	98.8%
10	97.9%	97.9%	100.0%	100.0%	82.8%

**Table 1.** Blocking artifact detection results.

#### 4.1. Hiding Traces of Double JPEG Compression

Instead of hiding an initial application of JPEG compression, an image forger may wish to remove evidence of recompressing a previously JPEG compressed image. Such a scenario might arise if an image forger wishes to alter a previously compressed image, then save the altered image as a JPEG. This is not an unlikely scenario since most digital cameras store captured images as JPEGs by default.

Several methods have been proposed to detect recompression of JPEG compressed images, commonly known as double JPEG compression [4], [5]. These methods operate by identifying artifacts introduced into an image’s DCT coefficient histograms as a result of quantizing the DCT coefficients twice using different quantization step sizes. As was previously discussed, when an image undergoes its initial application of JPEG compression, each set of DCT coefficients are clustered around integer multiples of the quantization step size. If a different quantization step size is used for any DCT subband during the second application of JPEG compression, an unequal number of quantized DCT coefficients will fall into each new quantization interval. As a result, a periodic signal will appear to modulate the DCT coefficient distributions of a doubly JPEG compressed image.

If our anti-forensic DCT artifact removal technique is properly applied, an image forger can prevent double JPEG compression artifacts from occurring in a doubly compressed image, thus rendering double JPEG compression undetectable through these means. To do this, anti-forensic dither must be added to the DCT coefficients of an image after the initial application of JPEG compression but before the image is compressed a second time. By doing this, the distribution of the anti-forensically modified DCT coefficients will match their distribution before the first instance of compression was applied, as was shown in [9]. Because of this, the DCT coefficient distributions will match those of an image that has only been compressed once after the image has been compressed a second time.

Fig. 3 shows an example demonstrating our anti-forensic technique’s ability to prevent the occurrence of double JPEG compression

artifacts. The leftmost plot shows the histogram of (3,3) DCT coefficients from an image that has been compressed once with a quality factor of 85. The center plot shows the histogram of (3,3) DCT coefficients from the same image after it has been doubly compressed, first with a quality factor of 75 then again with a quality factor of 85. In this plot, we can clearly see the presence of double JPEG compression artifacts. The rightmost plot shows the (3,3) DCT coefficient histogram from the image after it has been compressed once using a quality factor of 75, followed by the application of anti-forensic dither to the image’s DCT coefficients, then compressed again with a quality factor of 85. In this histogram, no double compression artifacts are present.

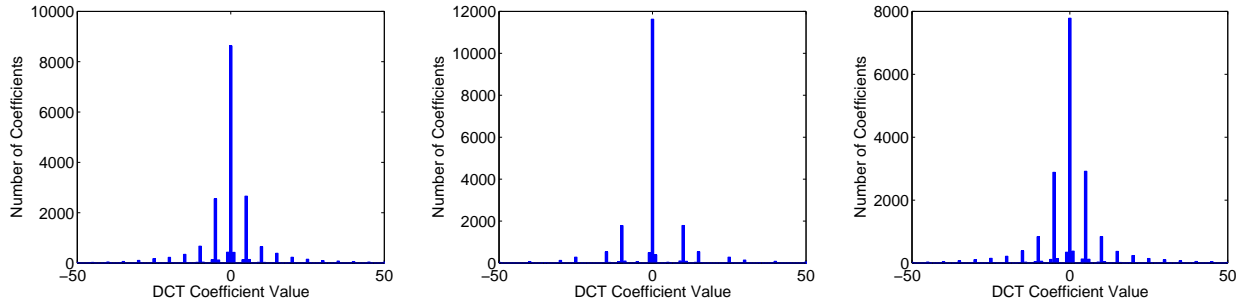
#### 4.2. Falsifying an Image’s Origin

In some scenarios, an image forger may wish to falsify the origin of a digital image. Simply altering the metadata tags associated with an image’s originating device is insufficient to accomplish this because several origin identifying features are intrinsically contained within a digital image. If an image has been JPEG compressed, one such feature is its quantization table. Because most digital cameras and image editing software use proprietary quantization tables when performing JPEG compression, the quantization tables used to JPEG compress an image can be used to help determine an image’s originating device [3]. Additionally, if the quantization tables match those used by image editing software, it suggests that the image may have been modified.

We are able to falsify this aspect of an image’s origin by first adding anti-forensic dither to an image’s DCT coefficients, then recompressing the image using quantization tables associated with another device. By modifying an image in this manner, we are able to insert the quantization signature associated with a different camera into an image while preventing the occurrence of double JPEG compression artifacts that may alert forensic investigators of such a forgery.

To demonstrate this, 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). Next, we applied anti-forensic dither to each image in the database, then separately recompressed each anti-forensically modified image using the quantization tables associated with each of the other cameras. We then obtained an estimate  $\hat{Q}_{i,j}$  of the quantization tables used to compress each of the recompressed images using the algorithm proposed in [1]. Each image was matched to a camera by choosing the camera whose quantization table  $Q_{i,j}^{(k)}$  resulted in the largest value of the similarity measure

$$s_k = \sum_i \sum_j \mathbb{1}(\hat{Q}_{i,j}, Q_{i,j}^{(k)}), \quad (3)$$



**Fig. 3.** 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).

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%	-

**Table 2.** Camera origin forgery classification results.

where  $\mathbb{1}(\cdot)$  denotes the indicator function.

Table 2 shows classification results from our camera forgery test. We were able to alter each image so that its compression signature was matched to the desired target camera with a 100% success rate in every case except when images captured by the Sony Cybershot DSC-V1 were falsified as images from the Sony Cybershot DSC-W80. In this case, only one image was not matched to the target camera, resulting in a 99% success rate.

### 4.3. Disguising Cut-and-Paste Forgeries

The detection of cut-and-paste forgeries, or inauthentic images created by cutting an object from one image and inserting it into another, is of particular importance to forensic researchers. Because these forgeries alter the content of an image, they can be used to falsify information in legal, intelligence, scientific and many other settings. Existing forensic techniques have been designed to detect cut-and-paste forgeries by identifying localized DCT quantization artifact discrepancies, such as the presence of single or double JPEG compression artifacts in one region of an image but not another [6]. Other techniques search for misalignments in JPEG blocking artifacts within an image [7].

In this paper and [9], we have shown that the DCT coefficient quantization artifacts and JPEG blocking artifacts that these forensic techniques rely upon for forgery detection can be removed from an image. This suggests that if anti-forensic dither and anti-forensic deblocking are applied to both images used when creating a cut-and-paste forgery, the forgery will be undetectable by these means. As a result, existing cut-and-paste forgery detection methods that rely upon JPEG compression artifacts cannot be trusted to verify that an image has not been altered.

## 5. CONCLUSIONS

In this paper, we have proposed an anti-forensic deblocking operation capable of reliably removing statistical traces of JPEG blocking

artifacts in images previously compressed using a quality factor of 30 or higher. Using this operation along with the anti-forensic operation that we proposed in [9], we have shown that it is possible to represent a previously JPEG compressed image as never-compressed, hide evidence of double JPEG compression, and falsify an image’s origin. Furthermore, these results suggest that our proposed anti-forensic operations can be used to create cut-and-paste image forgeries capable of avoiding detection by several forensic techniques that make use of JPEG compression artifacts.

## 6. REFERENCES

- [1] Z. Fan and R. de Queiroz, “Identification of bitmap compression history: JPEG detection and quantizer estimation,” *IEEE Trans. Image Processing*, vol. 12, no. 2, pp. 230–235, Feb 2003.
- [2] W. S. Lin, S. K. Tjoa, H. V. Zhao, and K. J. Ray Liu, “Digital image source coder forensics via intrinsic fingerprints,” *IEEE Trans. Information Forensics and Security*, vol. 4, no. 3, pp. 460–475, Sept. 2009.
- [3] H. Farid, “Digital image ballistics from JPEG quantization,” Tech. Rep. TR2006-583, Dept. of Computer Science, Dartmouth College, 2006.
- [4] A.C. Popescu and H. Farid, “Statistical tools for digital forensics,” in *6th International Workshop on Information Hiding*, Toronto, Canada, 2004.
- [5] T. Pevny and J. Fridrich, “Detection of double-compression in JPEG images for applications in steganography,” *IEEE Trans. Information Forensics and Security*, vol. 3, no. 2, pp. 247–258, June 2008.
- [6] J. He, Z. Lin, L. Wang, and X. Tang, “Detecting doctored JPEG images via dct coefficient analysis,” in *Proc. of ECCV*, 2006, vol. 3593, pp. 423–435.
- [7] S. Ye, Q.n Sun, and E.-C. Chang, “Detecting digital image forgeries by measuring inconsistencies of blocking artifact,” in *Proc. of ICME*, 2007, pp. 12–15.
- [8] M. Kirchner and R. Bohme, “Hiding traces of resampling in digital images,” *IEEE Trans. Information Forensics and Security*, vol. 3, no. 4, pp. 582–592, Dec. 2008.
- [9] M. C. Stamm, S. K. Tjoa, W. S. Lin, and K.J.R. Liu, “Anti-forensics of JPEG compression,” in *Proc. ICASSP*, Mar. 2010.
- [10] G. Zhai, W. Zhang, X. Yang, W. Lin, and Y. Xu, “Efficient image deblocking based on postfiltering in shifted windows,” *IEEE Trans. Circuits and Systems for Video Technology*, vol. 18, no. 1, pp. 122–126, Jan. 2008.
- [11] A.W.-C. Liew and H. Yan, “Blocking artifacts suppression in block-coded images using overcomplete wavelet representation,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 14, no. 4, pp. 450–461, April 2004.
- [12] G. Schaefer and M. Stich, “UCID: an uncompressed color image database,” in *Proc. SPIE: Storage and Retrieval Methods and Applications for Multimedia*, 2003, vol. 5307, pp. 472–480.