TRANSFORM CODER CLASSIFICATION FOR DIGITAL IMAGE FORENSICS

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

Dept. of Electrical and Computer Engineering, University of Maryland
College Park, MD 20742 USA

ABSTRACT

The area of non-intrusive forensic analysis has found many applications in the area of digital imaging. One unexplored area is the identification of source coding in digital images. In other words, given a digital image, can we identify which compression scheme was used, if any? This paper focuses on the aspect of transform coder classification, where we wish to determine which transform was used during compression. This scheme analyzes the histograms of coefficient subbands to determine the nature of the transform method. By obtaining the distance between the obtained histogram and the estimate of the original histogram, we can determine if the image was compressed using the transform tested. Results show that this method can successfully classify compression by transform as well as detect whether any compression has occurred at all in an image.

***Index Terms*—** Image and video forensics, image coding, source coding, transform coding, pattern classification.

1. INTRODUCTION

Non-intrusive forensic analysis is a relatively new research area which provides methods for extracting information from an output signal when the input (host) signal is unavailable. In the context of digital images, we use non-intrusive forensic analysis to identify operations such as blurring, sharpening, resizing, rotation, luminance adjustment, gamma correction, and more. Non-intrusive forensic analysis differs from traditional approaches to multimedia security – cryptography, video scrambling, watermarking, channel coding, etc. – which protect content using additive operations. For example, watermarking embeds a signal imperceptibly such that the additive signal is robust and traceable. In order to add the watermark, we require access to the original host signal. Similarly, with channel coding, we inject redundancy into a source in order to make it more robust in the presence of channel transmission errors. However, in many scenarios, we may not even have access to the host signal, and therefore we cannot enforce protection through any extrinsic means. With non-intrusive forensic analysis, the forensic analyst only has access to an output signal in a raw format, without any header information or metadata. Past operations performed upon the signal leave artifacts which become an intrinsic part of the signal, much like a fingerprint. We can analyze these artifacts to identify the history of operations.

Existing work in non-intrusive forensics for digital images primarily focuses on areas such as forgery detection, parameter estimation, image classification, and component forensics. In this paper, we address a new problem: the *identification* of source coding in images. In other words, given a previously-compressed image converted back into a raw format, what source coding scheme was used, or to what family of coders does it belong? What transform method, if any, was used? Discrete cosine transform (DCT)? Discrete wavelet transform (DWT)? Other? What wavelet basis was used, if any? Finally, how confident is our estimate?

Identification of source coding components and parameters has many applications in multimedia security, coding, and communication, especially when we lack access to the original signal or the device. One typical application of non-intrusive forensic analysis involves *patent infringement*. Often we wish to determine the specific encoding mechanism used within a broad category of source coders to detect potential patent infringement. This service is essential for detecting infringement in software and hardware products that are distributed for profit. By analyzing the artifacts that lossy coders leave behind, we can tell which coder was used along with its parameters.

Another application involves the verification of *digital image integrity*. The integrity of a digital image's content is of paramount importance in many forensic scenarios. The Scientific Working Group on Imaging Technologies – part of the International Association for Identification, an organization devoted to forensic science – cites potential legal ramifications regarding the use of image compression [1], which is often an unavoidable step in the image acquisition process. For example, the compression history of an image may become relevant in judicial proceedings, since one could argue that compression artifacts had obscured relevant information. Unfortunately, the compression algorithm and settings may not be immediately obvious, especially if performed automatically as a result of the acquisition device (e.g. compression in digital cameras). In this case, is there any way to determine the compression algorithm? This information is critical in subsequent quantitative image analysis, where the use of image compression can degrade the accuracy of object measurements. Such inaccuracies could lead to an incorrect diagnosis from a medical image, or an incorrect statement of guilt regarding a subject involved in a crime as viewed by a surveillance camera. Through non-intrusive forensic analysis, we can identify the nature of the compression module in the absence of the original image, thereby offering some measure of confidence regarding subsequent image analysis.

The first step in our forensic methodology for source coder identification involves the detection of pre-processing, namely block processing, which we address in an earlier work [2]. By estimating the block size, we identify the minimal unit upon which quantization is performed. In this paper, we address the next step in our methodology: *classification* of the transform method. (For our purposes, we restrict our discussion to those source coders which are most commonly used in practice, namely transform coders.) First, we will examine the nature of artifacts caused by a variety of transform coders. By discriminating amongst these artifacts, we can identify which source coder was used. Next, we present our method for transform coder classification, followed by results and a discussion.

Email: {kiemyang, wylin, kjrliu} @ umd.edu.

2. TRANSFORM COEFFICIENT CHARACTERISTICS

The main approach taken by a non-intrusive forensic system is the analysis of artifacts produced by the processing modules, and therefore we must first find the artifacts' source in transform coders.

In conducting non-intrusive forensic analysis, we must ask ourselves the following fundamental question. *For each source coder, where, and in what domain, does loss occur?* Consider the generic transform coder in Fig. 1 consisting of a 2-D transform, quantizer, and entropy coder. We see here that loss occurs during quantization and after the transform. Therefore, in order to conduct our forensic analysis, we must repeat the transform to return to the stage where loss occurs and examine the effect of quantization on transform coefficients.

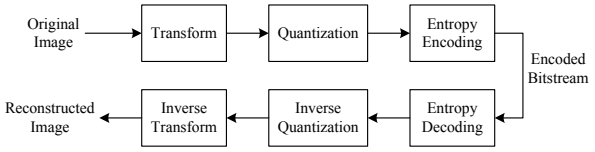

Fig. 1. Generic transform coding for digital images.

First, consider a DCT block coder. During quantization, the DCT coefficients are discretized. During inverse quantization, the quantized coefficients are multiplied by the quantization step size. As a result, we expect to find peaks in the histogram at multiples of the step size and zeros elsewhere. However, as noted by Fan and de Queiroz [3], due to the truncation and rounding effects caused during reconstruction, the histogram peaks do not appear as perfect impulses, as shown in Fig. 2.

Next, consider an embedded DWT coder. In an embedded zerotree coder such as EZW or SPIHT, the values of the transform coefficients are progressively transmitted by bit plane. Each embedded coder has its own algorithm for deciding the order in which the zerotree is traversed and the coefficients are transmitted. Since the coefficient values are bit-plane encoded, the transform coefficient histogram of the previously-compressed image will contain peaks at the designated reconstruction values. However, these reconstruction values are not spaced evenly, therefore the DWT coefficient histogram peaks are not periodic like that of the DCT. As with DCT block coding, rounding and truncation errors will add small imperfections to these histogram peaks. Fig. 2 shows the coefficient histogram at frequency (0,1) of a JPEG-coded image with quality factor of 70 and the coefficient histogram at the level-4 LH subband of a SPIHT-coded image with a bit rate of 1.0 bit per pixel.

How can we relate the generation of artifacts in DCT coding with that of embedded DWT coding? We can draw some insight from Xiong *et. al.* [4], which mentions that in a DCT block coder, each 8-by-8 block of transform coefficients can be treated as a 64-subband decomposition of the original 8-by-8 image block. In other words, we can treat the entire set of (0,0) coefficients as one subband, we treat all (0,1) coefficients as another subband, and so on. After tiling all of these subbands together, we obtain a coefficient subband representation similar to the one shown in Fig. 3. In this figure, a discrete cosine transformation with block size of 4 was applied to the image *Lena*. All of the DCT coefficients of the same frequency are combined into one subband, and these subbands are tiled together. The subband histograms are shown as well. Therefore, we can state that for both types of transforms, if we perform the appropriate subband decomposition and then observe the histogram

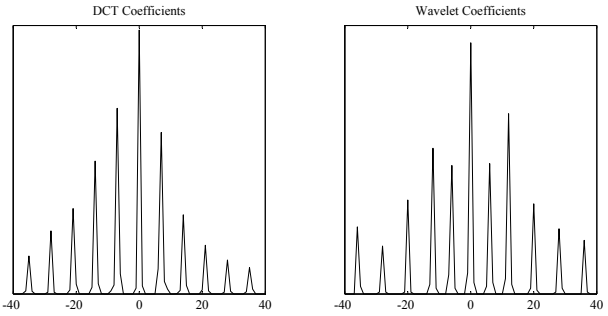

Fig. 2. Example coefficient histograms of two images previously compressed with different schemes. Left: DCT coefficient histogram of position (0,1) after JPEG decoding. Right: Wavelet coefficient histogram of the level-4 LH subband after SPIHT decoding.

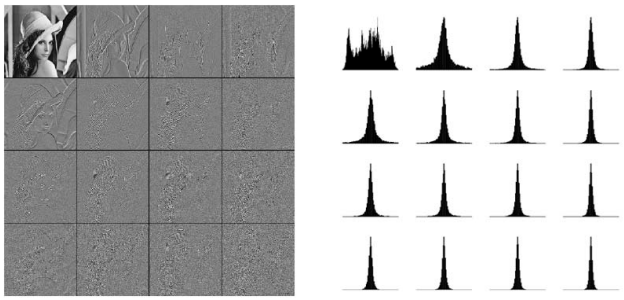

Fig. 3. Left: Reorganization of DCT coefficients into subbands. Right: Histograms for each coefficient subband.

within each subband, we should find histogram peaks. In fact, we can even apply the same concept to JPEG2000 images which have been coded using tiling. JPEG2000 allows the use of optional tile sizes of almost any size; subsequent coefficient transformation and quantization is performed on each tile separately. Nevertheless, if we obtain the wavelet coefficient subbands from each tile and then reorganize them as mentioned earlier, we arrive at a single coherent wavelet decomposition! This unifying concept shall become useful when formulating a transform classification scheme.

3. TRANSFORM METHOD CLASSIFICATION

Having discussed a major difference between the transform coefficient histograms of compressed and umcompressed images – the presence of histogram peaks – we would like to characterize this difference, perhaps using some sort of distance metric. In other words, after computing the histograms of transform coefficients, how similar are these histograms to the ideal transform coefficient histograms for a compressed or uncompressed image? Here, we propose a method which discriminates amongst coefficient histograms produced by different source coders, thereby achieving classification and identification of the compression scheme.

Unfortunately, to do this, we need the exact coefficient histogram before quantization, which is irretrievable. However, we can approximate the original coefficient histogram using a least-squares approximation. Research has previously shown that the histograms of DCT coefficients and wavelet coefficients are both accurately modeled using a generalized Gaussian distribution [5] [6] [7]. Therefore, let

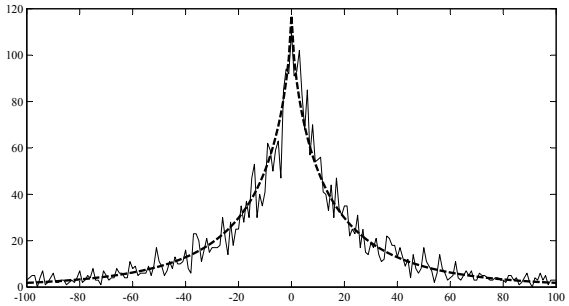

Fig. 4. Coefficient histogram from an uncompressed image, along with the nonlinear least-squares curve fit.

$p_i(k)$ be the probability mass function of the original coefficients within a subband, modeled as follows for simplicity:

$$p_i(k) = \gamma \exp(\lambda |k|^\nu) \tag{1}$$

where i is the index of the subband, γ is a normalization constant, $\lambda < 0$, and ν is the same exponent found in the generalized Gaussian distribution. One simple way to estimate the original coefficient histogram is through linear least squares. By taking the logarithm of $p_i(k)$, we can construct an over-determined linear system of equations, and we can solve for $\log \gamma$ and λ using the normal equations. When computing this least-squares solution, one may choose any suitable value for ν. For example, by fixing $\nu = 1$, we find the best fit to the Laplacian distribution.

Using linear least-squares to obtain a best fit for any exponential-based distribution is ill-advised, particularly due to sensitivity. Small perturbations in the histogram will yield widely different solutions. Furthermore, we are constrained to fixing ν constant, so we cannot really obtain the best fit for the generalized Gaussian distribution. To overcome these problems, we will use a nonlinear least-squares method to obtain the best fit. The optimization problem is formulated as follows:

$$\begin{array}{ll} \min_{\gamma,\lambda,\nu} & \sum_k \left(p(k) - \gamma \exp(\lambda |k|^\nu)\right)^2 \\ \text{s.t.} & \gamma > 0, \lambda < 0, \nu > 0 \end{array} \tag{2}$$

There are many numerical optimization methods to solve this sort of problem. We use a modified Newton method [8] which calculates the Newton direction using a modified Hessian which approaches the true Hessian, and we use a backtracking line search. Furthermore, we incorporate the linear equality constraints on γ, λ, and ν into the optimization problem by using the log-barrier function. Additional bounds on these variables is suggested and can be incorporated through additional barrier functions. This method of exponential curve fitting is relatively fast and offers a very high-quality model of coefficient histograms. Fig. 4 illustrates the performance of this estimate for an AC coefficient histogram with a bin width of 0.5 from an uncompressed image transformed using the DCT.

Now we can assess how well the histogram of inverse-quantized coefficients matches the least-squares estimate of the original histogram. With these two histograms, we can employ a probability-based distance metric. Let $\hat{p}_i(k)$ be the histogram of the coefficients in subband i from the reconstructed image. Consider the relative entropy, or Kullback-Leibler divergence, between the two distributions $\hat{p}(k)$ and $p(k)$.

$$D(\hat{p}||p) = \sum_k \hat{p}(k) \log \frac{\hat{p}(k)}{p(k)} \tag{3}$$

This distance metric, though not a proper distance metric in the strict sense, has a nice interpretation in our source coder identification problem. Intuitively, the relative entropy represents the extra number of bits required to encode a source with distribution $\hat{p}$ when provided a code for distribution p [9].

$$E_{\hat{p}} \left[\log \frac{1}{p} \right] = H(\hat{p}) + D(\hat{p}||p) \tag{4}$$

Therefore, if $D(\hat{p}||p)$ is low, then that indicates that the minimum number of bits required to represent $\hat{C}(i,j)$ is close to the number of bits required to represent $C(i,j)$. In other words, the image which we are testing is most likely uncompressed. If $D(\hat{p}||p)$ is high, then $\hat{C}(i,j)$ can be represented with far fewer bits then $C(i,j)$, and the tested image is most likely compressed. We compute $D(\hat{p}||p)$ for all coefficient subbands for which sufficient information exists (i.e. not completely quantized to zero). Our final distance metric is the median of all of these relative entropy values in the case of block transforms, or a weighted mean in the case of wavelet transforms. For the wavelet transform, we weight the relative entropy value for each subband by the size of the subband before averaging. This weighting guarantees equal contributions from all frequencies in the final distance measure.

The complete transform method identification algorithm is summarized as follows.

1. Choose a transform to test (e.g. DCT, Hadamard, DWT with Haar basis, etc.). Transform the image. Obtain the subband representation of the coefficients.
2. For each coefficient subband, obtain the histogram. (If insufficient information exists, move on to the next subband.)
3. Approximate the histogram of the original, unquantized coefficients using nonlinear least-squares estimation.
4. Calculate the relative entropy between the observed and the approximated original histogram.
5. Take the median value (for block transforms) or weighted mean (for wavelet transforms) of the relative entropies from all subbands for which sufficient information exists. This value is the final distance measure. If this value is high, then the transform method tested is the one used during compression.

4. RESULTS

First, we show the effectiveness of our method in discriminating between images which have been previously compressed versus images which have not been compressed at all. In Fig. 5, the four starred plots show the distance measure for four different images which have been compressed using JPEG (i.e. a DCT with block size of 8) for quality factors between 20 and 97. The four circled plots at the bottom represent the distance measure for images which have not undergone any compression at all, and therefore they do not change as a function of the quality factor. Note how there is a clear distinction between the two sets of lines, even for a quality factor as high as 97. This separation is achieved thanks to the quality of the nonlinear least-squares estimate of the original coefficient histograms. When least-squares estimation tries to approximate a histogram of coefficients from a previously-compressed image using a generalized Gaussian distribution, the curve fit will be poor and the KL divergence value will be high.

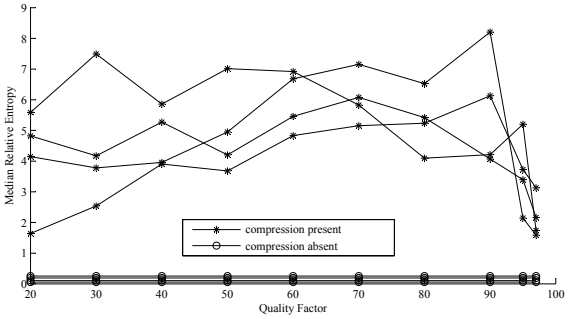

Fig. 5. Median relative entropy as a function of quality factor for four images when compression is present and absent.

	DCT	Hadamard	Slant	5/3	9/7	17/11
DCT	**2.198**	1.099	1.631	0.160	0.159	0.148
Hadamard	0.745	**2.208**	1.619	0.274	0.158	0.147
Slant	**2.805**	1.563	**2.249**	0.212	0.154	0.156
5/3	0.147	0.091	0.137	**4.099**	0.664	0.402
9/7	0.174	0.081	0.123	0.784	**4.186**	1.902
17/11	0.191	0.081	0.120	0.454	1.869	**4.216**

Table 1. Distance measures for six transforms, averaged over seven images. Rows headers indicate the true transform. Column headers indicate the tested transform. Values greater than 2.0 are highlighted.

Next, to illustrate the effectiveness of transform classification, we show a matrix of distance measures for various transforms averaged over seven standard test images. In Table 1, each row represents the true transform which was used during compression, and each column represents the transform for which the image is being tested. There are six transforms represented in total. The three block transforms – DCT, Hadamard, and slant – use a block size of 8, and baseline JPEG is used for quantization with a quality factor of 60. The three wavelet transforms – 5/3, 9/7, and 17/11 – use the standard five-level multiresolution decomposition, and SPIHT is used for quantization with a bit rate of 0.8 bits per pixel. Other quantization methods, such as the embedded block coding in JPEG2000, naturally produce similar results because histogram peaks remain present regardless of the quantization method.

In most cases, a high degree of separability is found between correct transform classification and incorrect classification. In particular, we see that classifying a block transform as a wavelet transform is very unlikely, and vice versa. Differentiating among the wavelet transforms is also successful as seen from the high level of separability among the distance measures. Differentiating among the block transforms is also successful, however the separability level among the distance measures is lower. One notable result from Table 1 is that many images which were compressed using a slant transform were classified as transformed using the DCT. This result is interesting, and in fact, it illuminates a key similarity between the two transforms. In particular, the transform matrices for the DCT and slant transforms similar in nature when the basis vectors are sequency ordered. In other words, if D_1 is the transform matrix for the DCT, and D_2 is the sequency-ordered transform matrix for the slant transform, then $D_1^T D_2$ is approximately equal to the identity matrix. Through this result, this scheme has illuminated an important, yet obscure, similarity by classifying these two transforms into one category, thus achieving the primary purpose of *classification*.

5. CONCLUSION

We have proposed a non-intrusive forensic procedure to classify the transform method used during compression in digital images. Given an image, which may or may not have been compressed using a transform coder, this method first obtains a subband representation of the transform coefficients. For each subband, the histogram of the original, unquantized transform coefficients is estimated using a nonlinear least-squares method. This numerical optimization method results in a high-quality curve fit for the generalized Gaussian distribution, unlike the linear least-squares method which must fix the exponent in the generalized Gaussian distribution and is more sensitive to perturbations in the input data. After calculating the relative entropy between the obtained histogram and the estimated original histogram for each subband, we arrive at a final distance measure; if this measure is high, then we classify the transform tested as being the true transform used during compression. As shown in the results, this method succeeds in distinguishing between images which have been previously compressed and those which have not. More importantly, this method succeeds in classifying the transform used during compression among six different transforms.

This entire process fits into a broader scheme for source coder identification, a new research topic which involves non-intrusive forensic analysis for digital images. The ultimate goal is an identification system which can determine the exact details of the compression method used (if any) upon a digital image. The benefits of such a system are significant, including the detection of patent infringement and verification of digital image integrity. Also, information gathered from a source coder identification system can potentially improve existing methods used in image quality assessment, rate control, and image restoration. As research in this area progresses, we believe that even more uses for a source coder identification system, and non-intrusive forensic analysis in general, will become apparent.

6. REFERENCES

[1] "Section 5: Recommendations and guidelines for the use of digital image processing in the criminal justice system," Tech. Rep., Scientific Working Group on Imaging Technology (SWGIT), International Association for Identification, Jan. 2006.

[2] Steven Tjoa, W. Sabrina Lin, H. Vicky Zhao, and K. J. Ray Liu, "Block size forensic analysis in digital images," in *ICASSP 2007*, Apr. 2007, vol. 1, pp. I–633–I–636.

[3] Zhigang Fan and Ricardo L. de Queiroz, "Identification of bitmap compression history: JPEG detection and quantizer estimation," *IEEE Transactions on Image Processing*, vol. 12, no. 2, pp. 230–235, Feb. 2003.

[4] Zixiang Xiong, Onur G. Guleryuz, and Michael T. Orchard, "A DCT-based embedded image coder," *IEEE Signal Processing Letters*, vol. 3, no. 11, pp. 289–290, Nov. 1996.

[5] Edmund Y. Lam and Joseph W. Goodman, "A mathematical analysis of the DCT coefficients distributions for images," *IEEE Transactions on Image Processing*, vol. 9, no. 10, pp. 1661–1666, Oct. 2000.

[6] Stephane G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693, July 1989.

[7] F. Muller, "Distribution shape of two-dimensional DCT coefficients of natural images," *Electronics Letters*, vol. 29, pp. 1935–1936, Oct. 1993.

[8] Stephen G. Nash and Ariela Sofer, *Linear and Nonlinear Programming*, McGraw-Hill, 1996.

[9] Thomas M. Cover and Joy A. Thomas, *Elements of Information Theory*, Wiley, 1991.