A Minimum Distortion Data Hiding Technique for Compressed Images

Çağatay Candan - Nikil Jayant
Multimedia Communications Laboratory
Georgia Institute of Technology, Atlanta, GA.
{candan,jayant}@ece.gatech.edu

Abstract— **We present a blind data hiding method for JPEG compressed images which minimizes the perceptual distortion due to data embedding. The proposed system presents a number of options to the encoder to cast the given hidden bits in the compressed content signal. The perceptual distortion cost of each option is calculated from the parameters available to the encoder such as the original image, quantization error due to compression and the Just Noticeable Distortion (JND) levels of the original image derived through an empirical human visual system model. The encoder selects the option with the minimum JND cost to cast the hidden bits. By the definition of blind decoding, the decoder should be able to extract the hidden bits without any side information on the option selected or the parameters available to the encoder. The decoder of the proposed system uses simple binary addition on the received transform coefficients to extract the hidden bits blindly. System performance is examined by computer experiments at different compression levels and at different embedding bitrates.**

I. Introduction

The goal of data hiding is embedding an imperceptible message into a carrier signal. The most well known application area of data hiding is the copyright protection / justification application, in which a copyright message is placed into the content signal. The desired features and the challenges of this application have been outlined in many places, [1], [2]. Some benchmarking tools have been developed to compare the performance of different approaches [3], [4]. In spite of some significant achievements in algorithm design [6], [7]; a universally accepted data hiding method meeting the high expectations of this particular application area has not been designed until now. We believe that this is mainly due to the inherent "battle of wits" situation between embedder and attacker (cryptographic aspects of the problem) and also partially due to the shortcomings at our understanding of some cognitive tasks (feature extraction / recognition of significant parts).

The subject of this paper lies in another application for data hiding. Our aim in here is to develop a blind hiding algorithm for compressed images operating at the least amount of embedding distortion. The blindness requirement imposes a fixed-system structure independent of the input signal. The goal of embedding data at low embedding distortion forces the system to make use of the local features of the input image. In this paper, we propose an algorithm which is fixed in structure yet adapts to the input signal so that a competition for minimum distortion embedding is established.

A possible application for this kind of data hiding method can be video captioning where the embedded caption messages travel within the data part of the video stream. Some resilience to the transmission noise and to light transcoding operation are the major requirements for this application.

Since the battle of wits situation of the copyright protection is eliminated for these applications, the main challenge shifts towards the signal processing aspects of the embedding operation. The management/diffusion of embedding distortion, system blindness, scalibility of embedding rate, compliance with existing compression systems are the main issues of interest. Furthermore similar to the image compression research, we can easily compare the performance of different hiding algorithms proposed for this application area by comparing the embedding distortion (say in PSNR) they produce.

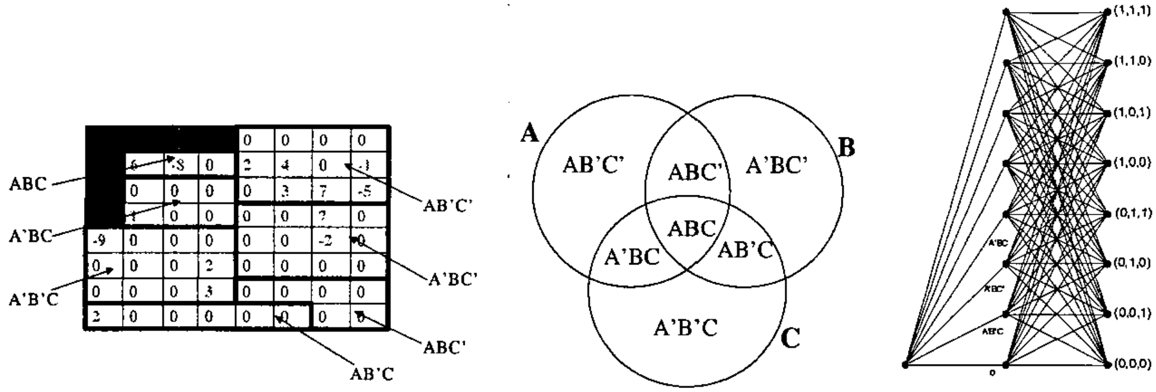
In this paper, we propose a blind data hiding method for JPEG compressed images which seeks a minimally distortive way of embedding the hidden data. The algorithm is essentially an extension of [5] which lies the groundwork of the approach taken in here. In this paper, we do not discuss any attack resilience issues. We believe that attack resilience problems should be tackled independently on a case-by-case basis, after the establishment of efficient means of embedding.

II. Proposed Data Hiding Method

There are two main approaches for blind data embedding. The first approach is the additive embedding approach which explicitly sums the hidden signal with the cover signal at a domain to form the marked signal [6]. The second approach is based on quantization. Quantization codewords (levels/locations) are grouped into two sets indicating the hidden bits of 0 and 1. The embedder quantizes the original pixel/coefficient value to the nearest element of the set of a given hidden bit [8].

To be compliant with the JPEG compression algorithm and to have some control on embedding distortion levels, we propose to use a DCT-domain quantization approach. JPEG compression quantizes the DCT coefficients of 8×8 blocks of the input image with a given quantization matrix which depends on user-desired bitrate/image quality. The algorithm proposed in here modifies the quantizated transform coefficients of the compressed image to produce a JPEG decodable image with the hidden data. The encoder uses the original (uncompressed) image and Watson's human visual system to decide on the best strategy to embed the hidden bits. Watson's human visual system model is used to facilitate the information about local image features into the hiding system. The effect of local image features (such as contrast masking, component masking) on transform coefficients are estimated through this model [9]. The decoder of the proposed system extracts the hidden bits blindly without the knowledge of original image and exact location/value of modified coefficients.

0-7803-7713-3/02/$17.00 © 2002 IEEE



(a) An 8 × 8 Image Block (b) Set Partitioning (c) Search Trellis

Fig. 2. Partitioning of an 8 by 8 block into three intersecting sets as shown in the middle and the search trellis

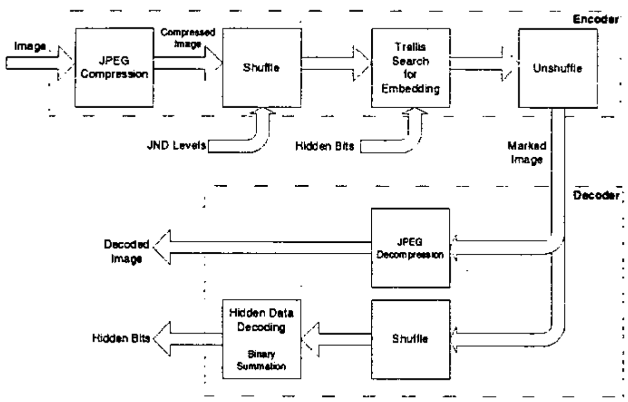


Fig. 1. The proposed encoder/decoder structure

The proposed method can be described as follows:

Step 0: The input image to be transmitted is compressed at the desired quality level by the JPEG compression system.

Step 1: Just Noticeable Distortion (JND) levels of the DCT-transform coefficients are estimated by the application Watson's Human Visual System (HVS) model [9].

Step 2: Quantized DCT coefficients are randomly shuffled to uniformly distribute the good and bad hiding opportunities throughout the image. The advantages of this operation is described further in the following paragraphs.

Step 3: Given the hidden information bitrate or the total number of bits to be embedded, a blocksize into which an integer number of bits can be embedded is determined. For example, if the hidden information bitrate is 5/64 bits/pixel, we may choose to embed 5 bits into each 8 by 8 blocks or 20 bits in 16 by 16 blocks.

Step 4: Randomly shuffled transform coefficients are collected into groups to embed an integer number of hidden bits per group.

Step 5: The proposed algorithm, which is described below, presents more than one option to embed the given bits per block. An exhaustive search is executed to find the embedding option with the least JND error. The options/search algorithm is discussed in the following paragraphs.

Step 6: The modified JPEG transform coefficients are unshuffled and therefore moved to their original locations. Finally, the image with the hidden data is transmitted.

Details: The design of a system with many hiding options to embed a set of hidden bits is at the heart of the problem. We illustrate our proposal with the following example:

Assume that 3 bits per 8 by 8 is chosen to be embedded in an image. An 8-by 8 block of compressed image is shown in Figure 2.

First, the block is partitioned into three intersecting sets. The partitioning strategy is pre-determined and known by the encoder/decoder.

We start the system description with the hidden bit extraction function of the decoder. The decoder extracts three hidden bits by the binary summation of the transform coefficients in their respective sets. For example, the decoder extracts the hidden bits from the block in Figure 2 as $\{0, 0, 1\}$.

$$\begin{aligned}\text{Bit-A} &= \sum_{ABC} + \sum_{ABC'} + \sum_{AB'C} + \sum_{AB'C'} = 10 \equiv 0 \,(\text{mod}\, 2) \\ \text{Bit-B} &= \sum_{ABC} + \sum_{A'BC} + \sum_{ABC'} + \sum_{A'BC'} = 4 \equiv 0 \,(\text{mod}\, 2) \\ \text{Bit-C} &= \sum_{ABC} + \sum_{A'BC} + \sum_{AB'C} + \sum_{A'B'C} = -3 \equiv 1 \,(\text{mod}\, 2)\end{aligned} \tag{1}$$

If the encoder decides to insert $\{1, 1, 1\}$ in the same block, it needs to modify the values of Bit-A and Bit-B in equation 1 before the transmission. This change can be done in a number of ways: we may increase or decrease the value of one of the coefficient in the set ABC' by one. This move changes the parity of the first two hidden bit equations and leaves the third one as it is. Or we may change one of the coefficients in the sets $AB'C'$ and $A'BC'$ to do the same correction. The search algorithm compares the JND cost of possible moves and selects the one with the least JND cost.

At the first step of this search process, the set-leaders of respective sets are determined.

The set-leader is defined as the element of a set with the least JND distortion if its value is increased or decreased by one.

We give a simple example for the determination of the set-leader of set ABC: Let's assume that the JPEG quantization step sizes of all three elements of the set ABC are 5. As can be seen from Figure 2, the elements of the set ABC are $(6, -8, 0)$. Then, at the decoder these transform coefficients are dequantized to $(30, -40, 0)$. Let's assume that the original values of transform coefficients before quantization are $(32.49, -40.1, 0.1)$. From this information, we can say that the set-leader of the set ABC is the first element of the set. Since an increment in the value of the first element from 6 to 7, moves the decoding point of the first element to 35 and this leads to an error of 2.51 which is the least possible error among the similar moves in this set.

In this simplified example we have not discussed the JND error associated with the increment/decrement operation instead we have compared the algebraic difference between the before and after values of transform coefficient. For the proposed system, the algebraic difference is normalized by the JND level of that particular transform coefficient. For example, if the JND level of the set-leader of the previously described example is 2, the JND error is the division of algebraic error by the JND value which is $2.51/2 = 1.255$ JND levels. Note that the original image, quantized values and JND levels are all available to the encoder.

After the election of the set-leaders, the cost of different options to embed hidden bits is evaluated and the one with the least value is selected. For the given example, if the bits to be embedded are $\{1, 1, 1\}$. The possible options are: modify the set-leader of ABC', or the set-leaders of two of the sets $\{AB'C', A'BC'\}$, $\{AB'C, A'BC\}$ and $\{ABC, A'B'C\}$.

A trellis search can be systematically executed to find the option with the minimum cost. In Figure 2, a general trellis is shown for this purpose. The metric of each branch connecting two nodes is the JND error of the set-leader providing the correction from node to node.

Decoder operation:

Step 0: Decompression of the image by the conventional JPEG decoder.

Step 1: Extraction of the hidden data by the binary summation operation as in (1) after the duplication of the shuffling operation. See Figure 1 for the illustration.

Comments: Basically, the method modifies the data stream of the compressed signal according to the hidden bits to be inserted. As expected, the embedding distortion due to a fix number of hidden bits may differ significantly depending on the level of compression that is applied on the content signal. The proposed data hiding process can be visualized as the perturbation of quantized transform coefficients around a selected operating point (compression rate). Depending on the slope of Rate-Distortion curve at the operating point the embedding distortion may be high or low. The proposed algorithm with shuffling is suitable for lightly and moderately compressed images. We examine the performance of the proposed method at different operating conditions in the computer experiments section.

Two critical design stages of the proposed system are the shuffling and partitioning stages. In this paper, we have applied a random shuffling strategy to uniformly distribute the transform coefficients of the blocks with high activity (high JND values) throughout the image. Note that, the transform coefficients with high JND values present *good opportunities* for hiding, since the perceptual distortion (JND) due to embedding is smaller for these coefficients. By shuffling, the good transform coefficients are distributed uniformly throughout the image. Also, we have observed at computer simulations that without the shuffling operation the blocks of image with very few non-zero transform coefficients (smooth regions of the image) tend to have a visible distortion after embedding. By the application of shuffling, this effect is also mitigated to some extent.

The partitioning strategy has a direct impact on the final distortion value. The number of coefficients in a partition and the selection of which coefficient should be placed into which partition requires a careful analysis. The partitioning strategy in terms of distortion control is linked to quantization step sizes of the JPEG compression system. For JPEG compression, the quantization table is known both at the encoder and decoder, therefore a partitioning strategy based on quantization step-sizes can be deployed without violating the blindness condition.

In the example given above, we have described a simple strategy that allocates 12 transform coefficients to the sets that are not at the intersection of any sets, 6 transform coefficients to the sets at the intersection of two sets and 3 transform coefficients to the set $\{ABC\}$. We plan to do a rigorous analysis to determine a provably good partitioning strategy.

If the decoder at the receiving end of the channel lacks the hidden bit extraction feature; the decoded image, the image with the hidden data is displayed with a minimum embedding distortion. If the decoder has the hidden data extraction feature, we do the shuffling operation done at the encoder, and then extract the hidden bits by binary summation operation. The information about shuffling pattern can be transmitted without an additional side-channel as follows: The random shuffling patterns for different size images can either be stored at the lookup tables of the encoder/decoder (as done in turbo error correction coding, i.e. interleaver design) or the shuffling pattern can be generated from the part of the data-stream which is not modified by the data-embedding function, like Huffman tables, JPEG quantization table values, image size etc.

The idea of shuffling has been introduced before [10], [11], [12]. Similar to our aim, Dr. Wu uses shuffling to disperse the active blocks over the image. Dr. Alturki uses shuffling to provide system security and robustness to the system design.

III. Computer Experiments

We have implemented the algorithm described at several carrier compression levels and at several embedding bitrates.

In Table I, we can see the PSNR results for the 256x256 Lena image at different hidden data payloads and at different compression levels. We have also given the compression distortion at the first line of the table for comparison purposes. We have chosen to report the results in terms of PSNR in order to be compatible with the compression literature. The same error can also be expressed in terms of JND levels.

We conclude that at high bitrates (high QF) the embedding distortion introduced through the proposed algorithm remains

TABLE I

DATA HIDING AND COMPRESSION DISTORTION OF LENA IMAGE (IN PSNR) AT DIFFERENT OPERATING POINTS. QF IS THE QUALITY FACTOR OF THE JPEG COMPRESSION. HIDDEN DATA BITRATE IS IN BITS PER 8×8 BLOCK.

	QF=50	QF=60	QF=70	QF=80	QF=90
0 bit	33.85	34.75	36.03	38.02	41.80
1 bit	33.75	34.68	35.98	37.99	41.78
2 bits	33.55	34.52	35.87	37.92	41.75
3 bits	33.29	34.35	35.72	37.86	41.69
4 bits	33.06	34.17	35.63	37.74	41.63

Fig. 3. Illustration of the embedding distortion. Compressed images without/with the hidden data are shown respectively on the top and bottom row of the Figure. The embedding rate is 4 bits per 8 by 8 blocks.

negligible in comparison with the compression distortion. As the payload increases, the introduced distortion increases at a slow rate. For low bitrates (low QF), the embedding distortion is accumulating at a much faster rate, making the proposed algorithm less suitable for the aggressively compressed images. In subjective tests conducted, none of the subjects were able to distinguish a difference between the compressed image with or without the hidden data for the $QF = 80$ and 4 bits per 8×8 block case. And only two subjects out of ten picked up the embedding distortion for the $QF = 50$ case at the same embedding rate.

IV. CONCLUSIONS

In this paper, we have presented a data hiding method for minimum distortion data embedding. The algorithm proposed selects and modifies the advantageous transform coefficients in such a way that the decoder can blindly extract the hidden bits. The algorithm is compatible with JPEG standard and easily upward scalable in terms of hidden information bitrate.

The algorithm consists of three stages: The first stage calculates the JND levels of all transform coefficients. At the second stage, the algorithm selects some of the transform coefficients with low JND errors after embedding (set-leaders) as possible candidates for the signaling of the hidden bits. Next, a search over the candidate space is made to calculate the JND cost of different options to embed the given hidden bits. The one with the least cost is selected at the encoder. At the decoder, the hidden information is extracted blindly through a simple binary summation operation. The information on the JND levels or set-leaders or the particular option selected is not needed for decoding.

Two critical design blocks of the algorithm is the random-shuffling and set-partitioning blocks. Without shuffling, algorithm makes less use of busy regions of the image. We plan to examine more coordinated methods of shuffling without violating the blindness condition.

At computer simulations, we have seen that the proposed algorithm is working effectively at moderate to light compression ratios. For more aggressive compression modes, the use of randomness for the distortion dispersion becomes less significant. Some context based rules or some post-processing type ideas might be useful for more challenging operating conditions.

We plan to examine alternative methods of shuffling and partitioning in the future. An alternative way of shuffling can be interchanging the DCT coefficients at the same location of two randomly chosen blocks (shuffling in the DCT channels).

The partitioning operation directly affects the embedding distortion. We plan to incorporate the knowledge of JPEG quantization levels which are both available at the encoder and decoder in the selection of partitions. Determination of provably good partitioning strategies is our near term goal.

REFERENCES

[1] "Special issue on digital watermarking," *IEEE Signal Processing Magazine*, September 2000.
[2] F. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Information hiding - a survey," *Proc. IEEE*, vol. 87, pp. 1062–1078, 1999.
[3] F. Petitcolas, "Watermarking schemes evaluation," *IEEE Signal Processing Magazine*, pp. 58–64, 2000.
[4] S. Pereira, S. Voloshynovskiy, M. Madueño, S. Marchand-Maillet, and T. Pun, "Second generation benchmarking and application oriented evaluation," *Information Hiding Workshop III*, 2001.
[5] N. Jayant, "Hiding a source identifier within a signal." US Patent Number:5,826,227.
[6] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. Image Process.*, pp. 1673–1687, 1997.
[7] M. K. Mıhçak and R. Venkatesan, "Blind image watermarking via derivation and quantization of robust semi-global statistics," *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2002.
[8] B. Chen and G. Wornell, "Quantization index modulation: a class of provably good methods for digital watermarking and information embedding," *IEEE Trans. Information Theory*, vol. 47, pp. 1423–1443, 2001.
[9] A. Watson, "Perceptual optimization of DCT color quantization matrices," *Proc. IEEE Int. Conf. on Image Processing*, pp. 100–104, 1994.
[10] M. Wu and B. Liu, "Digital watermarking using shuffling," *Proc. IEEE Int. Conf. on Image Processing (ICIP)*, pp. 291–295, 1999.
[11] M. Wu, *Multimedia Data Hiding*. PhD thesis, Department of EE, Princeton University, 2001.
[12] F. Alturki, *Theory and Applications of Data Hiding in Still Images*. PhD thesis, School of Electrical Engin., Georgia Institute of Technology, 2001.
[13] B. Tao and B. Dickinson, "Adaptive watermarking in the DCT domain," *Proc. IEEE Int. Conf. Acoust. Speech Signal Proc.*, pp. 2985–2988, 1997.