Media Compression via Data Hiding

Bin Zhu Ahmed H. Tewfik
Department of Electrical and Computer Engineering, University of Minnesota
Minneapolis, MN 55455

Abstract

Data hiding procedures embed digital data in a host audio, image or video signal. The embedded data is perceptually and statistically invisible until extracted by the proper algorithm. In this paper, we investigate the use of data hiding for media compression. We propose a media compression scheme which applies data hiding procedures to "fold" a media signal into itself. Specifically, the media signal to be compressed is split into host and residual parts. The residual part is compressed and embedded into the host part. The host part is compressed with standard compression techniques and transmitted to the decoder. As a result, only part of the original signal is needed to be compressed, and the perceptual quality of the compressed signal is not sacrificed. Bit rate reduction by the proposed method is proportional to the redundancy and irrelevancy remaining in the compressed signal. The proposed scheme works since no current compression schemes can completely remove the redundancy and irrelevancy in a signal.

1 Introduction

Data hiding is a procedure that allows extra information such as text, hypertext, labels, etc. to be embedded directly into a host media signal by making small modifications to its values. Several near-transparent high bit rate data hiding schemes for audio, image or video signals have recently proposed by the authors [1, 2, 3]. In these data hiding approaches, human perceptual masking models are used to maximize the energy of the hidden data while maintaining perceptual invisibility for the hidden information. At the receiver side, embedded data can be extracted from the received signal according to the extracting rule. These data hiding methods are fairly robust to compression, among other manipulations on signals.

In this paper, data hiding procedures are modified and applied to media compression. The media signal can be audio, image, video, or combination of them or with text. In the proposed approach, a media signal is split into two parts: the host and residual parts. The host part contains main components of the signal, and the residual part contains less information. The residual part is compressed and embedded into the host part. The host part embedded with data is compressed with standard compression techniques and sent to the decoder. In the proposed scheme, the embedded data is actually *part of the data from the signal to be compressed.* As a result, we *fold* part of the signal into itself. The data hiding procedures used in the proposed method are modified from the generic data hiding methods [2, 3] to survive only the distortion caused by the designated compression of the host part while minimizing perceptual distortion by the embedding data.

The proposed scheme reduces the bit rate since only part of the signal needs to be compressed. The perceptual quality of the compressed signal is kept the same since the embedding procedure does not introduce any perceptual distortion. For example, In an experiment on the 256 by 256 image Lena which used JPEG [4] to compress the host part at quality 75%, the bit rate was reduced by 33.8% with about the same perceptual quality, as compared to the direct compression by JPEG on the image at the same quality 75%. The proposed scheme works since no current compression schemes can completely remove the redundancy and irrelevancy in a signal. The more the redundancy and irrelevancy remain in the compressed signal, the more the proposed scheme saves. Thus the proposed scheme saves more at a higher bit rate than a lower bit rate for the same compression method, and saves more for a less efficient compression scheme than a more efficient compression scheme.

2 Data Hiding Algorithm for Image Coding by Folding

Our data hiding algorithms embed data into audio, image or video signal without causing any perceptual distortion [1, 2, 3]. The data hiding procedures employ human perceptual masking models to maximize the energy of the embedded data without incurring any *perceptual distortion* to the host signal. The data hiding method used in the proposed scheme is based on linear projection and quantization

1058-6393/98 $10.00

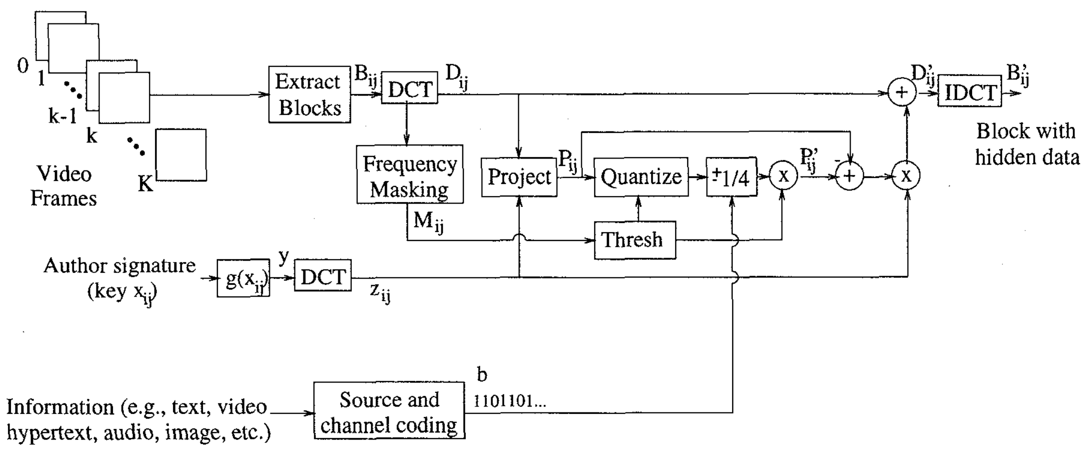

Figure 1. Diagram of video data hiding algorithm.

in the frequency domain. The block diagram for video data hiding is shown in Fig. 1. Each frame in a video sequence is first broken into small blocks B_{ij}, e.g., 8×8 or 4×4, and transformed into the DCT domain. Each block is represented as a vector $\vec{v}$ in an n-dimensional space, e.g., $n = 8^2$ or 4^2. This is shown in Fig. 2.

Each block $\vec{D}$ is projected onto a pseudo-random, author defined, direction $\vec{Z}$: $p =< \vec{D}, \vec{Z} >$. The projection p is then quantized to a masking-based threshold T of the block. The quantized value $p^* = round(P/T)$ is then perturbed by $(-1)^{b-1}\frac{1}{4}T$ to embed a bit b. The new projection $p' = p^* \pm \frac{1}{4}T$ contains the hidden data bit b. One way to achieve the required perturbation for the projection value p is to constrain the modification to be along the projection direction $\vec{Z}$. In this case, the new vector with the embedded data is represented by

$$\vec{D}' = \vec{D} + (p' - p)\vec{Z}. \quad (1)$$

The vector $\vec{D}'$ appears identical to $\vec{D}$, thanks to the masking values given by HVS, but now contains hidden information.

To recover the hidden bit, a receiver projects $\vec{D}'$ from the received video onto the random direction $\vec{Z}$. The projection value is quantized by the masking-based threshold. The integer portion is removed, and the reminder is used to determine whether the embedded bit is a 1 or 0:

$$b = \begin{cases} 1 & \text{if } (< \vec{D}', \vec{Z} > /T - \left[< \vec{D}', \vec{Z} > /T\right]) > 0 \\ 0 & \text{otherwise,} \end{cases} \quad (2)$$

The embedding equation Eq. 1 has the advantage that it can be easily extended into multiple projection directions per block. Thus more than one data bit can be embedded into an block. For example, to hide two bits per block, we use two orthonormal directions $\vec{Z}_1$ and $\vec{Z}_2$. Since the two projections are orthogonal to each other, perturbation along one direction does not affect the other. The perturbed vector can be simply expressed as:

$$\vec{I}' = \vec{I} + (p_1' - p_1)\vec{Z}_1 + (p_2' - p_2)\vec{Z}_2 \quad (3)$$

where $p_1 =< \vec{I}, \vec{Z}_1 >$, $p_2 =< \vec{I}, \vec{Z}_2 >$, and p_1' and p_2' are the perturbed projection values, found by the method described above. More general cases and details can be found in [3].

The procedures to embed data to host image and audio signals are similar to the video data hiding described above. Interested readers are referred to [3].

The data hiding method described above is a generic method, which means that the possible distortions to the host signal are unknown *a priori*. In media compression applications, the distortion during the coding process is known. In the case of JPEG coding, the distortion arises from a known quantization table. The data hiding method described previously is modified to survive only the designated quantization stage in the coding process. To do this, the host signal is first quantized and reconstructed by the quantization table at the designated quality. The reconstructed signal is used as the host signal to hide the data. The embedding procedure is also modified as follows. First, the perturbed projection remainder need not to be $\pm T/4$. In a general data hiding application, the distortion to a host signal is usually unknown priori, so it is necessary to set the perturbed projection remainder at $\pm T/4$ to maximize the safety margin for both data bit 0 and 1. In media compression applications the only distortion to the host signal is the quantization step in coding process, thereby the perturbed projection remainder can be relaxed to be any value in the range $(0, T/4)$ for $b = 1$ and any value in the range $(-T/4, 0)$ for $b = 0$. Second, the modification is not along the direction $\vec{Z}$. Each DCT coefficient is modified independently according to its visual masking value. However, the

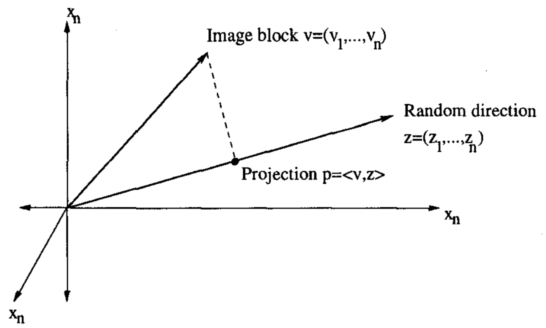

Figure 2. Diagram of block-based projection.

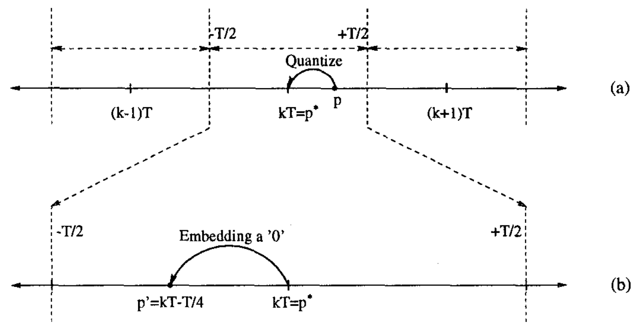

Figure 3. Diagram of embedding data, (a) quantization of projection by masking threshold T, and (b) embedding a '0' by perturbing the quantized value by $-\frac{T}{4}$.

projection onto $\vec{Z}$ after modification must be in the right range, determined by the hidden data bit. The modification of the DCT coefficients is done in such a way that it minimizes the perceptual errors between the modified signal block and the block of the original signal. To make the hidden bits free from errors after the compression of the host signal, we only need to make sure that the remainder of the perturbed values divided by the threshold along the $\vec{Z}$ direction, after compression quantization and reconstruction, still remain in the correct range.

3 Media Compression via Data Hiding

The proposed media compression via data hiding is shown in Fig. 4. At the first stage, the input signal is split into two parts: a host part and a residual part. The host part contains the most relevant components of the signal. It is used as the host signal to the residual part in the data hiding step. The residual part contains less important components. It is compressed and embedded into the host part. In this way, we *fold* part of an signal into the other part.

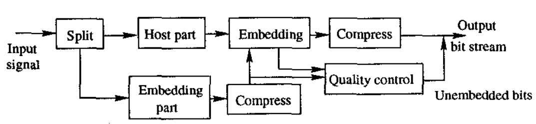

Figure 4. Block diagram of the proposed compression scheme.

For an image, it can be split by a subband decomposition. The low frequency band is used as the host part, while the high frequency band is used as the residual part. Similar method can be used for an audio signal. For a video clip, I frames act as the host part, and B frames are used as the residual part. P frames can be either as the host part or the residual part, depending on the compression configuration and bit rate. For composite signals such as the classroom video, shown in Fig. 8, which consists of video and speech of four different languages, it is naturally to use the video as the host part and the speech as the residual part.

Before embedded into the host part, the residual part is compressed. For images, a modified embedded zerotree wavelet coder [5] is used to compress the residual image. This compression method is chosen because it is very efficient in compression and fully embedded: it can stop at any bit rate without sacrificing its compression efficiency. For audio and video, MPEG compression is used.

The embedding procedure is described in the previous section. The host part embedded with the data from the residual part is then compressed to the designated quality by a traditional compression method. For host image, JPEG is used, for host audio or video, MPEG compression is used.

A host signal has a certain capacity to hold embedded data without errors. Sometimes the bits from the residual part is more than the capacity of the host part. In this case, we have to output the bits that have not been embedded into the host part. The quality control unit estimates the number of bits needed from the compressed residual part to match the perceptual quality of the host part, and outputs the bits, if there is any, that are not embedded into the host part.

Figure 5. Original 256 by 256 gray-scale image Lena.

4 Experimental Results

To illustrate the performance of the proposed media coding method, we have applied both the JPEG compression and the proposed method to the 256 by 256 gray-scale image Lena (8 bits per pixel). The original image is shown in Fig. 5. The 9-7 bi-orthogonal wavelet filters reported in [6] is used to split the original image. Fig. 6 shows the JPEG compressed image of the original image at JPEG quality 75%. The resulting image requires 87155 bits to encode. The image coded using our data hiding algorithm (with JPEG quality 75% on the host image) is shown in Fig. 7. The image is encoded using 57657 bits. The bit rate is reduced by 33.8%. Perceptual tests on a Sun Spark 5 computer show that the images shown in both Figs. 6 and 7 are about the same.

We have also applied the proposed method to the classroom video which consists of video and four channel speech of different languages, as shown in Fig. 8. The video consists of 250 grayscale frames of size 360×240. The speech streams are sampled at 8 kHz and represented with 8 bits per sample. A Code Excited Linear Predication (CELP) voice compression algorithm is applied to each speech signal. They are then embedded into the video sequence which is compressed by motion JPEG at compression ration 7 to 1, as shown in Fig. 8(b). At this compression ratio, no speech is needed to be transmitted: all of them can be embedded into the video without any errors. The recovered speech signals are shown in Fig. 8(c).

5 Conclusion

We have proposed a media method which employs the data hiding technology to fold one part into the other part of a signal. The signal to be compressed is split into a host image part and a residue image part. The residue part is compressed and embedded into the host image part. As a result, only the host portion needs to be compressed when stored or transmitted. Experimental results have shown that the method reduced bit rate while maintaining perceptual quality.

References

[1] Mitchell Swanson, Bin Zhu, and Ahmed H. Tewfik, "Transparent Robust Data Hiding in Images," *7th IEEE Digital Signal Processing Workshop*, PP. 37-40, 1996.

[2] Mitchell Swanson, Bin Zhu, and Ahmed H. Tewfik, "Data Hiding for Video-in-Video," to appear *1997 IEEE Int. Conf. on Image Proc.,* Santa Barbara, California, Oct. 27–29, 1997..

[3] B. Zhu, "Coding and Data Hiding for Multimedia," *Ph. D. Thesis,* University of Minnesota, 1997.

[4] G. K. Wallace, "The JPEG Still Picture Compression Standard," *Communications ACM*, Vol. 34, No. 4, PP 31-44, 1991.

[5] A. Said and W. A. Pearlman, "New, Fast, and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees," *IEEE Trans. Circuits & Syst. for Video Tech.,*. Vol. 6 No. 3, pp. 243–250, 1996.

[6] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image Coding Using the Wavelet Transform," *IEEE Trans. Image Processing,* Vol. 1, PP. 205-220, 1992.

Figure 6. Image compressed directly by the JPEG method at JPEG quality 75%.

Figure 7. Image compressed by the proposed coding method at JPEG quality 75%.

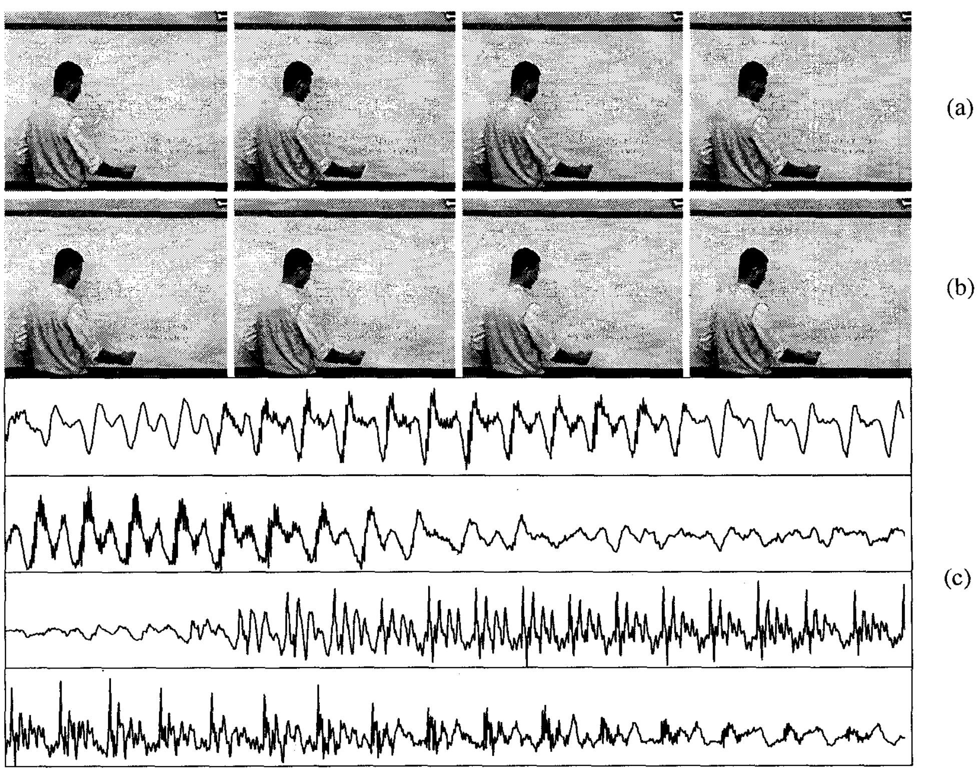

Figure 8. Classroom video sequence (a) Original video sequence, (b) Motion JPEG compressed at compression ratio 7:1 (c) The recovered four channel speech signals.