The Best Transform in the Replacement Coefficients and the Size of the Payload Relationship Sense

Eric A. Silva
Department of Electrical and Computer Engineering, Tufts University
Medford, MA, USA

Sos S. Agaian
Department of Electrical Engineering, University of Texas
San Antonio, TX, USA

Abstract

Steganography is the hiding of payload data within cover data. The goal of a steganographic method is to minimize the visually apparent and statistical differences between the cover data and a steganogram while maximizing the size of the payload. Current digital image steganography presents the challenge of hiding data in a digital image in a way that is robust to image manipulation and attack. Concealment of the payload data is contingent upon minimizing the apparent differences between the cover data and the stego data. There are two basic steganographic methods: Least significant bit (LSB) and transform-based. LSB steganography is one of the simplest methods. Data hidden in images using LSB steganography is highly sensitive to image alteration and vulnerable to attack. Transform based steganography is potentially more resistant to loss from image manipulation and increases the difficulty to a potential attacker. This paper describes a steganography implementation using the insignificant coefficients of a fast transform of an image for data hiding. The performance of this method using different transforms is measured and the characteristics of each transform are examined. Performance is measured by examining the payload density and its effect on image degradation. The performance of different transforms is measured on a variety of image types.

Introduction

Steganography is the hiding of *payload* data within *cover* data, so as to conceal the existence of the payload data. The combination of payload data and cover data is called a steganogram.

Steganography has a cousin in cryptography. The payload may be encrypted to secure its contents as unreadable. Steganography seeks not to make the contents of a message secret—which is cryptography's role—but to make secret the *existence* of a message. The two primary classes of digital image steganography are least significant bit steganography and transform based steganography. LSB steganography is more susceptible to image manipulation than transform based steganography. Transform based steganography has the potential to achieve higher payload capacity than LSB steganography.

The steganographic method described in this paper embeds data in the transform domain of an image. This method differs from other transform based steganographic methods in that instead of slightly modifying a large number of coefficients in either the transform or spatial domain, a small number of insignificant coefficients are altogether replaced. The method may be implemented using a number of different transforms. This paper describes the implementation using the discrete cosine transform (DCT), fast Fourier transform (FFT), Haar wavelet transform, and Hadamard transform. The four different implementations of the steganographic method yield varying performances and characteristics. The method in this paper is a first step in the creation of a steganographic algorithm that is robust to JPEG compression.

Transform-Based Steganographic Method

This section presents a transform (DCT, FFT, Hadamard transform, and Haar wavelet transform) based algorithm. Note that DCT is used in common image compression protocols such as JPEG and MPEG. Most images on the Internet are in the JPEG format, so this paper focuses on storing steganographic data within the image by altering the coefficients in the transform domain. The forward and inverse DCT for an N by N block are given by (1) and (2) respectively.

$$F_{uv} = \frac{C_u}{2}\frac{C_v}{2}\sum_{y=0}^{N-1}\sum_{x=0}^{N-1} f_{xy}\cos\left(\frac{(2y+1)v\pi}{2N}\right)\cos\left(\frac{(2x+1)u\pi}{2N}\right) \quad (1)$$

$$f_{xy} = \sum_{v=0}^{N-1}\sum_{u=0}^{N-1}\frac{C_u}{2}\frac{C_v}{2}F_{uv}\cos\left(\frac{(2y+1)v\pi}{2N}\right)\cos\left(\frac{(2x+1)u\pi}{2N}\right) \quad (2)$$

with

$$C_p = \frac{1}{\sqrt{2}} \quad \text{for } p = 0,$$
$$C_p = 1 \quad \text{for } p > 0$$

The definitions of the other transforms may be found in Ref. [4].

A step-by-step example of algorithm is described below in figure 1. The coefficients generated by the orthogonal transform are first divided into two classes: significant and insignificant. This classification is made by comparing the coefficient to the *significance threshold T*. The significance threshold is the value below which the transform coefficients will be deemed insignificant.

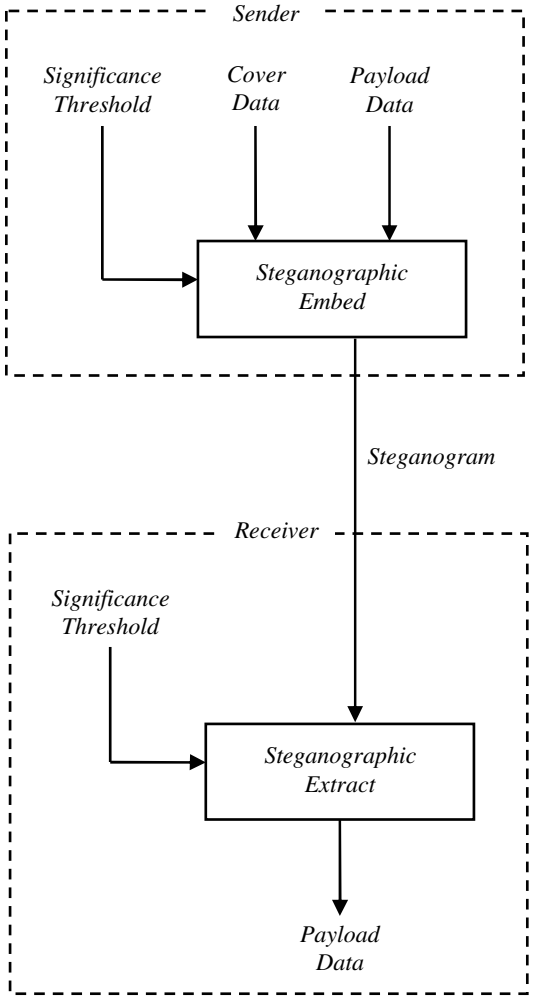

Figure 1. Algorithm Dataflow

This value is an input to the system and is chosen by the sender. Changing the values of significant coefficients would likely perceptibly alter the image; however, changing the values of insignificant coefficients would not. The coefficients deemed insignificant will be replaced with payload data. The payload data will by scaled by the *payload-scaling factor* $Sc_{Payload}$ shown by (3). With the payload data being 8-bit characters ranging in value from 0 to 255, the payload scaling factor is:

$$Sc_{Payload} = \frac{T}{2^8}\frac{1}{2} \quad (3)$$

$Sc_{Payload}$ is halved in (3) because of the payload shifting factor $Sh_{Payload}$ which is shown below in equation (4).

$$Sh_{Payload} = -T \quad (4)$$

Scaling the payload by the payload scaling factor and then shifting it in the negative direction an amount equal to the significance threshold ensures that the payload data replacing the insignificant coefficients remains within the range of insignificance. This is important because the significance of the transform coefficients in the steganogram image is used to determine which coefficients contain payload data and which do not. The original coefficients found to be insignificant are discarded. The replacement coefficient R_{uv} is calculated in (5) to contain an element P_i of the payload data.

$$R_{uv} = P_i Sc_{Payoad} + Sh_{Payload} \quad (5)$$

Naturally, the payload and the significance threshold should be transmitted over different communications channels. The extraction algorithm is easily inferable from the embedding algorithm. The receiver first calculates the transform of the steganogram. The receiver then classifies each transform coefficient into significant or insignificant. The receiver knows the significance threshold T; therefore she can calculate the payload scaling factor $Sc_{Payload}$ as well as the payload shifting factor $Sh_{Payload}$. Each insignificant coefficient is shifted then scaled (reversing the order taken by the sender) to yield a fragment of the payload data. The entire payload will be generated in this way.

The payload is shifted by the significance threshold to distribute the scaled payload across the entire range of insignificance—both positive and negative. An alternate method to distribute the payload in this way is to treat positive and negative values separately, replacing positive insignificant coefficients with positive, scaled replacement coefficients (containing a fragment of the payload) and doing the same for negative coefficients. As with the encryption of payload data, the payload may also be processed before embedding to evenly distribute its values across the entire range of possible values. The following is a synopsis of the embedding algorithm (taking the cover data, payload data,

significance threshold, and transform—DCT, FFT, Haar, or Hadamard—as parameters):

A Synopsis of the Embedding Algorithm

1. Payload data, cover data, and significance threshold are input
2. Scale each member of the payload data by $Sc_{Payload}$
3. Shift each member in the negative direction by $Sh_{Payload}$
4. Perform the transform of choice on the cover data
5. Classify each transform coefficient as significant or insignificant based upon the significance threshold
6. Index each insignificant transform coefficient and discard them
7. In a logical order agreed upon by the sender and receiver replace each discarded insignificant transform coefficient with a replacement coefficient (5)
8. Perform the inverse transform on the new set of coefficients to generate the steganogram
9. Output steganogram

Experimental Details

Measurement of the performance of different transforms for embedding data in images is performed by embedding data below varying significance thresholds, and subsequently varying densities, and measuring image degradation. Degradation is measured by mean squared error (MSE) between the cover and the steganogram. *Payload density* is the fraction of transform coefficients that are discarded and subsequently replaced with payload data.

Several cover images and four transforms were chosen to measure performance. Each of the four transforms, DCT, FFT, Hadamard transform, and Haar wavelet transform, is used to embed data in each cover image. Figure 3 through figure 8 show images and charts of the payload densities versus image degradation for each sample image.

The DCT, FFT, Hadamard transform, and Haar wavelet transform are used to embed data in the cover image. The difference between the steganogram and cover data is calculated for each instance of embedding. The mean squared error of the difference is calculated to measure image degradation due to changes made embedding the payload data.

As is shown in figure 1, the sender must provide the algorithm with a cover image, payload data, and a significance threshold *T*. The sender must agree upon the ordering of the payload within the coefficients (e.g. columns then rows.) All coefficients below the significance threshold are discarded.

The significance threshold chosen by the sender is partly dependent upon the size of the payload. Clearly, as the significance threshold increases, the more coefficients are discarded and replaced with payload data. This contributes to image degradation and the likelihood that the payload will be discovered. The transform used also highly influences the significance threshold. This is because different transforms yield widely varying scales. For example, for the image *Paris.tif* shown in figure 7, the magnitudes of coefficients vary from 3.81×10^{4} to 2.32×10^{-4} in the DCT domain while the magnitudes of coefficients are only as high as 4.90×10^{2} in the Haar wavelet transform domain—different orders of magnitude. Care must be taken in choosing a significance threshold with regard to the transform used. The optimal significance threshold will perfectly fit the payload within steganogram.

The coefficients are altogether replaced with those containing the payload data. To make the data extractable (by keeping the payload data within the significance threshold) and to avoid significantly changing the visual characteristics of the image, the payload data must be scaled to within the threshold.

Changing the values of significant coefficients would likely perceptibly alter the image; however changing the values of insignificant coefficients would not. The replacement coefficients are based on the payload data and on the range of insignificance, not at all on the original, discarded coefficients.

Effects of Significance Threshold

The relationship between the chosen significance threshold and the size of the payload a given cover image will accommodate (primary axis) can be seen in figure 2. In addition, the MSE was calculated for the same set of significance thresholds (secondary axis). The relationship between significance threshold, payload capacity, and image degradation is illustrated below. The image used was that of figure 8, *CapeCod.tif*.

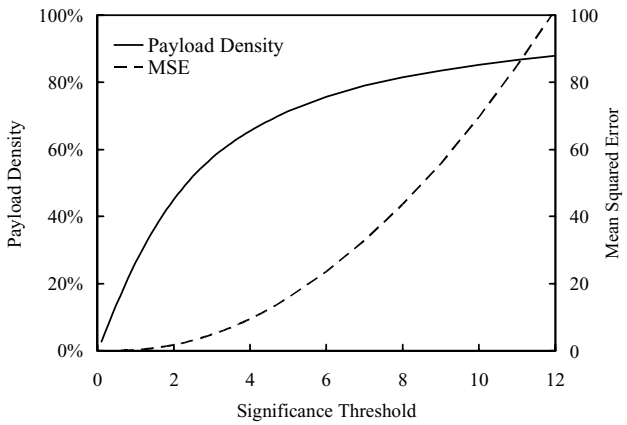

Figure 2. – Effects of Significance Threshold on Payload Density and Mean Squared Error for the DCT

Figure 3. Sample Cover Image Barcelon.tif

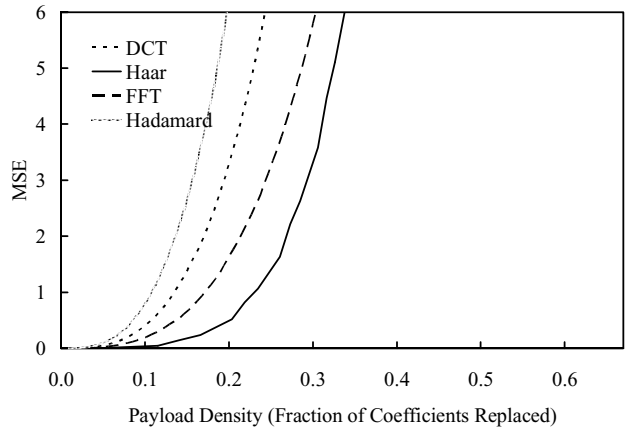

Figure 4. Barcelon.tif: Payload Density's Effect on Image Degradation

Figure 5. Sample Cover Image CapeCod.tif

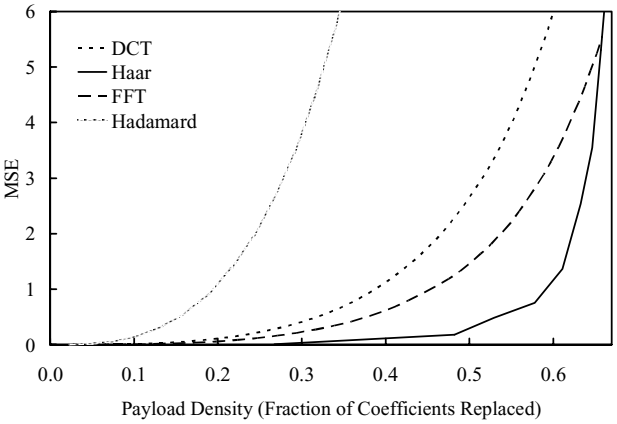

Figure 6. Paris.tif: Payload Density's Effect on Image Degradation

Figure 7. Sample Cover Image Paris.tif

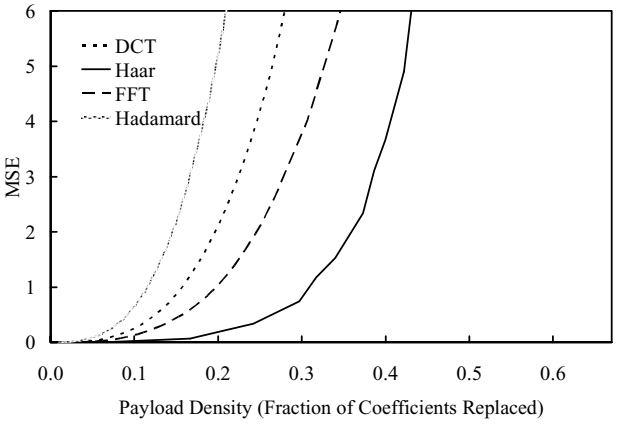

Figure 8. Paris.tif: Payload Density's Effect on Image Degradation

Results

The relative performances of the transforms are uniform across each of the three sample images. The absolute performance of the steganographic method varied significantly from image to image.

The embedding data using this steganographic method achieves the highest payload density with the lowest induced MSE when data is embedded in the Haar wavelet transform domain. Images that are visually less complex offer greater capacity and less degradation. This effect can be seen in when comparing the charts in figures 4, 6, and 8. The simplest of the images, *CapeCod.tif* shown in figure 4, exhibited the least amount of image degradation at a given payload density—for all transforms.

The performances of the transforms in the steganographic method are uniform across the sample images. Their performances are ordered as such:

1. Haar Wavelet Transform
2. FFT
3. DCT
4. Hadamard Transform

An interesting characteristic of the Haar wavelet transform is the flatness of its curve in figure 6, a very simple image. A large fraction (50%) of the coefficients of the image in figure 5 are discarded and replaced with payload data while keeping the MSE below .5. Other simple images yield similar results when using this steganographic method with the Haar wavelet transform.

Conclusion

We have embedded data in different transform domains while inducing little image degradation with this new approach. The performance model consists of measurement of the changes produced when embedding data. The Haar wavelet transform proved to be the best choice for this method. In fact, the relative performances of each of the transforms used were uniform across all images tested. Future work will seek to expand upon this method to hide data in the discrete cosine transform or wavelet domain in compressed images.

References

1. Jessica Fridrich, Miroslav Goljan, and Dorin Hogea, Steganalysis of JPEG Images: Breaking the F5 Algorithm, *5th Information Hiding Workshop*, Noordwijkerhout, The Netherlands, pp. 310-323 (2002).
2. Andreas Westfeld, F5-A Steganographic Algorithm: High Capacity Despite Better Steganalysis, *4th International Workshop on Information Hiding* (2001).
3. *Webster's Revised Unabridged Dictionary*, MICRA, Inc., (1998).
4. Alexander D. Poularikas, editor, Richard C. Dorf, editor-in-chief, *The Handbook of Transforms and Applications*, CRC Press, Boca Raton, FL, (1995).
5. Sos Agaian, Benjamin Rodriguez, Glenn Dietricht, Steganalysis Using Modified Pixel Comparison and Complexity Measure, *SPIE Conference,* (2004).
6. Johnson, N.F., Jajodia, S., *Steganalysis of images created using current steganographic tools*, http://www.ise.gmu.edu/~njohnson/ihws98/jjgmu.html, (1998).
7. R. Chandramouli, Nasir Memon, *Analysis of LSB based Image Steganography Techniques*, *IEEE*, p. 1019-1022, (2001).
8. Westfeld, A. Pfitzmann, *Attacks on Steganographic Systems*, Lecture Notes in Computer Science, vol. 1768, Springer-Verlag, Berlin, 2002, p. 61-75

Biographies

Eric A. Silva is a firmware engineer at RSA Security Inc. in Bedford, MA and is currently a Masters candidate in the Department of Electrical and Computer Engineering at Tufts University. He holds a Bachelors degree in computer systems engineering from the University of Massachusetts at Amherst.

Sos S. Agaian is Full Professor, College of Engineering, The University of Texas at San Antonio and an Adjunct Professor in the Department of Electrical Engineering, Tufts University, MA. He has authored more than 250 scientific papers, three books, and holds 12 patents. He is an associate editor of the Journal of Real-Time Imaging, the Journal of Electronic Imaging, and an editorial board member of the Journal Pattern Recognition and Image Analysis. His current research interests lie in the broad area of Signal/image processing with applications, Information security, and Quantum signal processing, and communication.