High Capacity Steganographic Method Based Upon JPEG

Adel Almohammad
Brunel University
adel.almohammad@brunel.ac.uk

Robert M. Hierons
Brunel University
rob.hierons@brunel.ac.uk

Gheorghita Ghinea
Brunel University
george.ghinea@brunel.ac.uk

Abstract

The two most important aspects of any image-based steganographic system are the quality of the stego-image and the capacity of the cover image. This paper proposes a novel and high capacity steganographic approach based on Discrete Cosine Transformation (DCT) and JPEG compression. JPEG technique divides the input image into non-overlapping blocks of 8x8 pixels and uses the DCT transformation. However, our proposed method divides the cover image into non-overlapping blocks of 16x16 pixels. For each quantized DCT block, the least two-significant bits (2-LSBs) of each middle frequency coefficient are modified to embed two secret bits. Our aim is to investigate the data hiding efficiency using larger blocks for JPEG compression. Our experiment result shows that the proposed approach can provide a higher information-hiding capacity than Jpeg-Jsteg and Chang et al. methods based on the conventional blocks of 8x8 pixels. Furthermore, the produced stego-images are almost identical to the original cover images.

1. Introduction

Message transmissions over the Internet still have data security problem. Therefore, secure and secret communication methods are needed for transmitting messages over the Internet. Cryptography scrambles the message so that it cannot be understood. However, it makes the message suspicious enough to attract eavesdropper's attention. Steganography hides the secret message within other innocuous-looking cover files (i.e. images, music and video files) so that it cannot be observed.

Steganography aims to hide the very existence of communication by embedding messages within other cover objects. However, watermarking aims to protect the copyrights of digital media (i.e. images, music, video and software) owners. Therefore, the goal of steganography is the secret messages while the goal of watermarking is the cover object itself [3].

There are two kinds of image steganographic techniques: spatial domain and frequency domain based methods. The schemes of the first kind directly embed the secret data within the pixels of the cover image such as Least Significant Bit (LSB) insertion. The schemes of the second kind embed the secret data within the cover image that has been transformed such as DCT (discrete cosine transformation). The DCT coefficients of the transformed cover image will be quantized, and then modified according to the secret data [6].

The capacity, the amount of data embedded within a given image, of spatial domain schemes is better than that of the second kind. However, the frequency domain schemes have better robustness than that of the first kind [14]. Since the watermarking schemes require to be robust, most of the watermarking schemes used are frequency domain ones in order to protect the ownership of target images. However, steganographic methods do not need to be robust and instead the capacity and quality (imperceptibility) are important. Many novel embedding techniques have been suggested in order to enhance the security, increase the capacity, and improve the robustness of steganographic methods [3, 6-8, 12].

One of the most popular image formats widely used on the Internet and World Wide Web (WWW) is JPEG. JPEG compression provides large compression ratio and maintains high image quality. The DCT transform has been adopted by the Joint Photographic Experts Group (JPEG). Moreover, JPEG has been widely used within the steganographic community as a cover image [10-13].

In the JPEG compression, as stated in JPEG standard, the image is divided into disjoint blocks of 8x8 pixels, a 2-dimensional DCT is applied to each block, and then the DCT coefficients of these blocks are quantized and coded [4]. Most of the steganographic techniques used for JPEG images adopt the standard JPEG compression. The cover image is divided into non-overlapping blocks of 8x8 pixels in order to perform DCT and provide compressed images [2, 7, 8, 10-13].

A third party may suspect in some stego-images and may detect the existence of a hidden message within such images. In order to avoid this, the design of

0-7695-3102-4/08 $25.00 © 2008 IEEE
DOI 10.1109/ARES.2008.72

IEEE computer society

Authorized licensed use limited to: Brunel Universit. Downloaded on June 21, 2009 at 05:31 from IEEE Xplore. Restrictions apply.

almost all the steganographic systems takes into account the characteristics of the human vision system (HVS). The HVS is more sensitive to lower frequency noise. Therefore, the low frequency coefficients should be avoided when hiding secret messages in order to achieve better imperceptibility of hidden messages. However, the high frequency coefficients will be discarded after the quantization process of compression. This is because the energy of natural images is concentrated on the lower frequency components [2]. Therefore, the middle frequencies will be the most appropriate region for embedding and it will be used in our proposed scheme.

The rest of this paper is organized as follows. Section 2 will review the related work on JPEG steganographic methods. Section 3 will propose our data hiding scheme based upon JPEG. Section 4 will show our experiment result and will discuss the attributes of our proposed method. Finally, the conclusion will be presented in Section 5.

2. Related work

The DCT calculation for block-sizes larger than 8x8 pixels may require much more running time and may increase the computational operations and complexity [1]. This might be one of the reasons why the standard JPEG uses blocks of 8x8 pixels.

Tseng and Chang proposed a novel steganographic method based on JPEG. The DCT for each block of 8x8 pixels was applied in order to improve the capacity and control the compression ratio [12].

The widely known JPEG-based steganographic tool Jpeg-Jsteg divides the cover image into non-overlapping blocks of 8x8 pixels. It embeds the secret data in the LSB of the quantized DCT coefficients of each block. Since it embeds only one bit in each quantized coefficient whose value is not 1, 0, or -1, the capacity of this method is very limited [14].

Since the energy of images is concentrated in the lower frequency coefficients, modifying such coefficients may cause a quality degradation of output image. However, high frequency coefficients will be discarded due to the quantization process. Chang et al. developed a steganographic method based upon JPEG and modified 8x8 quantization table in order to improve the hiding capacity of Jpeg-Jsteg method. They utilized the middle frequency for embedding in order to achieve better hiding capacity and acceptable stego-image quality [2].

In related work, Bracamonte et al. used disjoint blocks of 16x16 pixels in order to improve the JPEG compression. They found that there just four coefficients had significant magnitude (low frequency coefficients). Therefore, only these four coefficients have to be calculated and this reduced the computational overhead significantly. They got a better compression ratio by using larger block-size for JPEG compression [1].

Increasing the capacity of cover images while maintaining imperceptibility is still a challenge. Since the significant DCT coefficients of 16x16-pixels blocks are limited, more middle frequency coefficients can be used for embedding. This might increase the embedding capacity and preserve image quality. We suggest a steganographic method based upon blocks of 16x16 pixels and modified 16x16 quantization table. Therefore, we are going to use the same technique used by Chang et al. However, we divide the cover image into non-overlapping blocks of 16x16 pixels and use larger quantization table in order to improve the embedding capacity. In order to verify the improvements our method may achieve, we compare it with both of Jpeg-Jsteg and Chang et al. methods.

3. The proposed method

As mentioned in the previous sections, almost all steganography research done in the JPEG transformation domain divides a given cover image into non-overlapping blocks of 8x8 pixels. Since the computational capabilities have improved significantly over the last decade, calculating the DCT for blocks of 16x16-pixels or larger may be much more feasible than before.

3.1. Quantization tables

The JPEG standard uses 8x8 quantization tables, but it does not specify default or standard values for quantization tables. Specifying the quantization values is left up to the application. However, the JPEG standard provides a pair of quantization tables (luminance and chrominance) as examples tested empirically and found to generate good results (Table.1 for luminance). Dividing this quantization table by (2), we get a new quantization table (Table.2). Using this new quantization table generates reconstructed images almost identical to the source image [4]. Therefore, this table will be used with Jpeg-Jsteg method in our experiment. Since the values of these tables could be an arbitrary choice [9], some researchers modified these quantization tables for their research purposes. For example (Table.3), the modified version of Table.2, has been used within Chang et al. method. 8x8 quantization tables apart, there are no samples for larger quantization tables in the JPEG standard. Miano states that “If you are implementing a JPEG encoder

Authorized licensed use limited to: Brunel Universit. Downloaded on June 21, 2009 at 05:31 from IEEE Xplore. Restrictions apply.

you can come up with your own scaling or use any other method you want for generating quantization values". Therefore, a quantization table can arbitrarily be generated [9]. Consequently, we produced a 16x16 quantization table (Table.4) by simulating and stretching the scaled quantization table (Table.2). For embedding purposes, the middle frequencies of the produced quantization table were set to be 1 in the Table.4. However, the process of generating the most appropriate quantization table for steganography is out of our scope of this paper and is left as a topic for future work.

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Table.1 The luminance quantization table (example in the JPEG standard)

8	6	5	8	12	20	26	31
6	6	7	10	13	29	30	28
7	7	8	12	20	29	35	28
7	9	11	15	26	44	40	31
9	11	19	28	34	55	52	39
12	18	28	32	41	52	57	46
25	32	39	44	52	61	60	51
36	46	48	49	56	50	52	50

Table.2 The scaled quantization table (Jpeg-Jsteg) (scale factor =2)

8	6	5	8	1	1	1	1
6	6	7	1	1	1	1	28
7	7	1	1	1	1	35	28
7	1	1	1	1	44	40	31
1	1	1	1	34	55	52	39
1	1	1	32	41	52	57	46
1	1	39	44	52	61	60	51
1	46	48	49	56	50	52	50

Table.3 The modified quantization table of Chang et al.

3.2. The embedding and extracting procedures

The procedure of embedding a secret message in a cover image for JPEG-based steganography is illustrated in the left side of Fig.1. It can be described as follows:

1. The message (M) to be embedded in the cover image is randomly generated.
2. The cover image is divided into non-overlapping blocks of 16x16 pixels and then the DCT is used to transform each block into DCT coefficients.
3. The DCT coefficients are scaled by the modified 16x16 quantization table (Table.4). In this quantization table, the values of (1) represent the middle frequencies to be used for embedding (121 values). The quantized DCT coefficients of each block are rounded to the nearest integers and then set in zigzag scan order.
4. The least two-significant bits of each middle frequency coefficient in the quantized DCT blocks are modified to embed two secret bits.
5. The JPEG entropy coding (DPCM, Run-Length coding, and Huffman coding) is applied to compress these resultant blocks, and then the JPEG file is obtained.

The procedure of extracting the embedded message from the JPEG file is illustrated in the right side of Fig.1. In the extracting procedure, the JPEG file (stego-image) is entropy decoded using the coding tables (Huffman tables) located in the image header. As a result we get the blocks of quantized DCT coefficients modified according to the secret message. From each pre-defined middle frequency coefficient of each block we retrieve the least two-significant bits (secret bits). We put these retrieved bits in the same order of embedding to get the secret message (M).

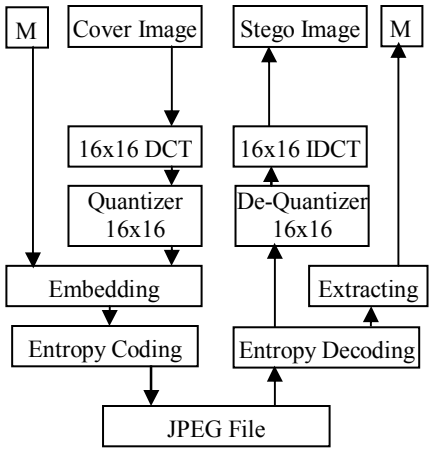

Fig.1 The block diagram of embedding (left) and extracting (right) procedures

4. Evaluation and discussion

We conducted some experiments in order to evaluate the efficiency of our method. Three gray-level Images: Sony, Lena, and Pepper (Fig.2 (A), Fig.3 (A) & Fig.4 (A) respectively), each of 160x160 pixels were used as cover images.

The three methods (Jpeg-Jsteg, Chang et al., and our proposed method) were coded in Matlab R2006b

Authorized licensed use limited to: Brunel Universit. Downloaded on June 21, 2009 at 05:31 from IEEE Xplore. Restrictions apply.

(V 7.3.0) and run on a PC Pentium 4 with 1GB of RAM under the Windows XP operation system.

Most researchers use Peak Signal to Noise Ratio (PSNR) and Mean Square Error (MSE) criteria to measure the quality of image coding and compression [5]. The PSNR and MSE for an NxN gray-level image are defined as [2]:

$$PSNR = 10.\log_{10} \frac{255^2}{MSE} dB \qquad (1)$$

$$MSE = \left(\frac{1}{N}\right)^2 \sum_{i=1}^{N} \sum_{j=1}^{N} \left(Xij - \overline{Xij}\right)^2 \qquad (2)$$

Xij : The pixel values of the cover image.

$\overline{Xij}$: The pixel values of the stego-image.

Table.5 shows the sizes of the stego-images (Kbytes) for three steganographic methods; Chang et al. method, Jpeg-Jsteg method, and our proposed method.

Table.6 shows the capacity (bits) of the cover images using these three methods. Jpeg-Jsteg method does not embed a fixed number of secret bits within a given cover image. However, the capacity varies from block to another, so this method includes code to count the number of secret bits embedded within each cover image. Chang et al. uses the 2-LSB of each predefined middle frequency coefficient (set to be 1 in Table.3) for embedding. Each block of 8x8 pixels can embed 2x26 secret bits. Therefore, a cover image of 160x160 pixels can hold 52x (160x160)/ (8x8) =20800 secret bits. However, our proposed method can embed 242 secret bits in each block of 16x16 pixels using the same technique (Table.4). Therefore, the capacity of a cover image of 160x160 pixels is 242x (160x160)/ (16x16) = 24200 secret bits.

Table.7 shows the stego-images' quality of these three methods. Our method provides stego-images with acceptable quality in comparison with other two methods.

Table.8 shows the computational time of the DCT transformation, quantization, and embedding phases required for each steganographic method. The running time of our method is a bit less than that of Chang et al. method. However, Jpeg-Jsteg requires less computation time than the other two methods because it embeds less secrets bits.

We can justify these results as following; comparing Table.3 with Table.4 we can notice that more high frequency coefficients will be discarded in our method (120 bottom right part of Table.4) than Chang et al. method (4x (28 bottom right part of Table.3)). Therefore, the size of the stego-images of our method should be smaller than Chang et al. method. However, the result of the experiment did not reveal this (Table.5). The main reason may be related to the generated quantization table which has to be selected and examined much more carefully. However, the header's size of the stego-images for our method was increased by 192 bytes ((16x16)-(8x8)) because of using larger quantization table. Therefore, using larger cover images may improve this result. The stego-images of our method have good quality because we kept a considerable number of coefficients to represent each block. Even though we have slightly modified the middle frequency coefficients, it gives better results than discarding such coefficients.

16	8	7	6	6	1	1	1	1	1	1	1	1	1	1	1
7	7	6	6	1	1	1	1	1	1	1	1	1	1	1	30
7	6	6	1	1	1	1	1	1	1	1	1	1	1	30	28
6	8	1	1	1	1	1	1	1	1	1	1	1	32	35	29
8	1	1	1	1	1	1	1	1	1	1	1	32	35	32	28
1	1	1	1	1	1	1	1	1	1	1	35	40	42	40	35
1	1	1	1	1	1	1	1	1	1	35	44	42	40	35	31
1	1	1	1	1	1	1	1	1	35	44	44	50	53	52	45
1	1	1	1	1	1	1	1	31	34	44	55	53	52	45	39
1	1	1	1	1	1	1	31	34	40	41	47	52	45	52	50
1	1	1	1	1	1	30	32	36	41	47	52	54	57	50	46
1	1	1	1	1	36	32	36	44	47	52	57	60	60	55	50
1	1	1	1	36	39	42	44	48	52	57	61	60	60	55	51
1	1	1	39	42	47	48	46	49	57	56	55	52	51	54	51
1	1	41	46	47	48	48	49	53	56	53	50	51	52	51	50
1	43	47	47	48	48	49	57	57	56	50	52	52	51	50	50

Table.4 Our suggested 16x16 quantization table.

Fig.2 (B), Fig.3 (B) & Fig.4 (B) show the stego-images of Chang et al. method, while Fig.2 (C), Fig.3 (C) & Fig.4 (C) show the stego-images of Jpeg-Jsteg method. We can compare the quality of these stego-

Authorized licensed use limited to: Brunel Universit. Downloaded on June 21, 2009 at 05:31 from IEEE Xplore. Restrictions apply.

images visually with our method's stego-images (Fig.2 (D), Fig.3 (D) & Fig.4 (D)). For HVS, they are almost identical and it is difficult to find out a difference among them. Even when we have the unmodified original-images; it is not easy to assure the difference between the original images and the stego ones.

The capacity and imperceptibility represent the two main requirements of any steganographic technique [15]. Our steganographic method provides larger embedding capacity and acceptable stego-image's quality.

Method/Image	Sony	Lena	Pepper
Original	26	26	26
Chang et al.	7.89	10.3	10.8
Jpeg-Jsteg	2.97	4.74	5.09
Our Method	8.57	11.33	11.9

Table.5 The sizes (KB) of the stego-images.

Method/Image	Sony	Lena	Pepper
Chang et al.	20800	20800	20800
Jpeg-Jsteg	2026	3394	3787
Our Method	24200	24200	24200

Table.6 The capacity (Bits) of the stego methods.

Method/Image	Sony	Lena	Pepper
Chang et al.	48.4	44.4	46.4
Jpeg-Jsteg	45.0	41.6	42.5
Our Method	48.0	44.1	46.5

Table.7 The quality (PSNR: db) of the stego images.

Method/Image	Sony	Lena	Pepper
Chang et al.	1.62	1.58	1.57
Jpeg-Jsteg	1.25	1.33	1.35
Our Method	1.56	1.56	1.56

Table.8 The computational time (sec) of the steganographic methods.

Almost always the color space YCbCr is used to store JPEG images. The component Y (luminance) represents the intensity of the image. However, the components Cb and Cr (chrominance) specify the blueness and redness of the image respectively. Using only the Y component in such color model (YCbCR) produces a gray-level representation of the color image [9]. Therefore, gray-level images represent special cases of color images. As a result, color images can be used as cover images but we have to take all of these components (Y, Cb, and Cr) into consideration. In this case, each of the chrominance components should have the same procedures that Y component have in Fig.1.

5. Conclusion

We proposed a novel steganographic method based upon JPEG compression and DCT transformation. Using gray-level cover images, we transformed (DCT) non-overlapping blocks of 16x16 pixels instead of non-overlapping blocks of 8x8 pixels. The transformed DCT coefficients were quantized by a modified 16x16 quantization table. Then, we embedded the secret data within the middle frequency coefficients.

In order to evaluate our method, we used the DCT transformations and the suggested 16x16 quantization table with Chang et al. embedding technique. Afterwards, we compared our method with Jpeg-Jsteg method and Chang et al. method to show the improvements provided by our method.

The results showed that our method can embed more secret data than the methods which use 8x8 blocks (Chang et al. method & Jpeg-Jsteg method). The stego-images of our method were almost identical to other methods' stego-images and it is difficult to differentiate between them and the original images. However, the size of the stego-images could be optimized when the quantization table is much more carefully selected.

6. References

[1] J. Bracamonte, M. Ansorge and F. Pellandini, "Adaptive block-size transform coding for image compression", *IEEE International Conference on Acoustics, Speech, and Signal Processing. ICASSP-97*, 21-24 April, Vol 4, 1997, pp. 2721-2724.

[2] C.-C. Chang, T.-S. Chen and L.-Z. Chung, "A steganographic method based upon JPEG and quantization table modification", *Information Sciences,* vol. 141, 2002, pp. 123-138.

[3] R. Chu, X. You, X. Kong and X. Ba, "A DCT-based image steganographic method resisting statistical attacks", *In Proceedings of (ICASSP '04), IEEE International Conference on Acoustics, Speech, and Signal Processing*, 17-21 May. vol.5, 2004, pp V-953-6.

[4] ISO DIS 10918-1 "Digital Compression and Coding of Continuous-Tone Still Images (JPEG)", *CCITT Recommendation T.81*.

[5] X. Kong, R. Chu, X. Ba, T. Zhang and D. Yang, "A Perception Evaluation Scheme for Steganography", *in Intelligent Data Engineering and Automated Learning*, vol. 2690: Springer Berlin / Heidelberg, LNCS, 2003, pp. 426-430.

[6] Y. K. Lee and L.-H. Chen, "High capacity image steganographic model", *Vision, Image and Signal Processing, IEE Proceedings*, June, 147(3), 2000, pp. 288-294.

[7] Y.-K. Lee and L.-H. Chen, "Secure Error-Free Steganography for JPEG Images", *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 17, 2003, pp. 967-981.

Authorized licensed use limited to: Brunel Universit. Downloaded on June 21, 2009 at 05:31 from IEEE Xplore. Restrictions apply.

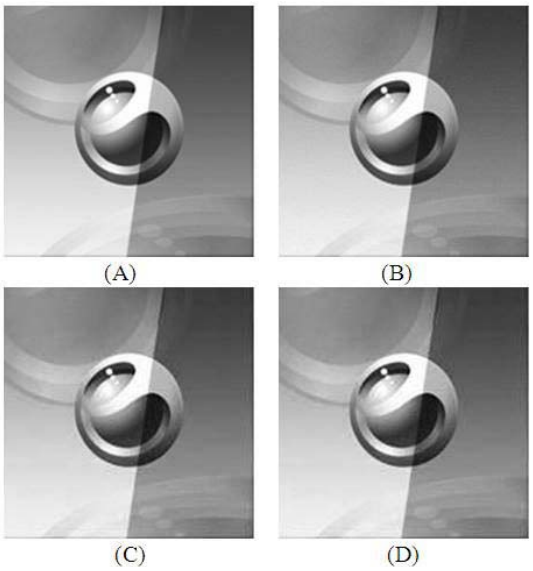

Fig.2 Sony image (A): original image, (B): stego-image of Chang et al. method, (C): stego-image of Jpeg-Jsteg method, (D): stego-image of our method.

Fig.3 Lena image (A): original image, (B): stego-image of Chang et al. method, (C): stego-image of Jpeg-Jsteg method, (D): stego-image of our method.

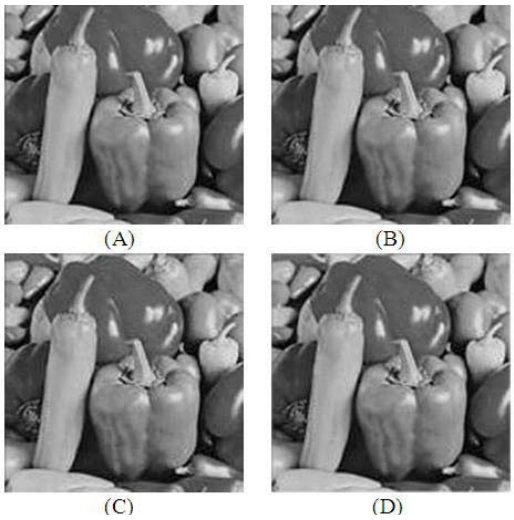

Fig.4 Pepper image (A): original image, (B): stego-image of Chang et al. method, (C): stego-image of Jpeg-Jsteg method, (D): stego-image of our method.

[8] Q. Li, C. Yu and D. Chu, "A Robust Image Hiding Method Based on Sign Embedding and Fuzzy Classification", *The Sixth World Congress on Intelligent Control and Automation, 2006. WCICA 2006*, June 21-23. 2006, pp.10050-10053.

[9] J. Miano, "Compressed Image File Formats: JPEG, PNG, GIF, XBM, BMP", Addison-Wesley, 1999.

[10] V. K. Munirajan, E. Cole and S. Ring, "Transform domain steganography detection using fuzzy inference systems", *In Proceedings of the IEEE Sixth International Symposium on Multimedia Software Engineering*, 13-15 Dec. 2004, pp. 286-291.

[11] J. Rongrong, Y. Hongxun, L. Shaohui, W. Liang and S. Jianchao, "A New Steganalysis Method for Adaptive Spread Spectrum Steganography", *In Proceedings of the International Conference on Intelligent Information Hiding and Multimedia Signal Processing, IIH-MSP '06*, Dec 2006, pp. 365-368.

[12] H.-W. Tseng and C.-C. Chang, "Steganography using JPEG-compressed images", *The Fourth International Conference on Computer and Information Technology, CIT '04*, 14-16 Sept 2004, pp. 12-17.

[13] P. H. W. Wong and J. W. C. Wong, "A Data Hiding Technique in JPEG Compressed Domain", *In Proceedings of SPIE Conference on Security and Watermarking of Multimedia Contents III*, San Joes, CA, USA, Jan 2001, vol. 4314, 2001, pp. 309-340.

[14] Y.-H. Yu, C.-C. Chang and Y.-C. Hu, "Hiding secret data in images via predictive coding", *Pattern Recognition*, vol. 38, 2005, pp. 691-705.

[15] X. Zhang and S. Wang, "Steganography using multiple-base notational system and human vision sensitivity", *Signal Processing Letters, IEEE*, vol. 12, 2005, pp. 67-70.

Authorized licensed use limited to: Brunel Universit. Downloaded on June 21, 2009 at 05:31 from IEEE Xplore. Restrictions apply.