Short Paper

A Robust Image Hiding Method Using Wavelet Technique*

CHIH-YANG LIN AND YU-TAI CHING
Department of Computer and Information Science
National Chiao Tung University
Hsinchu, 300 Taiwan

A robust wavelet-based image hiding methods, that hide still images, *E*, inside a covered image, *C*, to establish a composite image, *P*, are presented. We can hide up to three full-size embedded images inside a cover image while maintaining the quality of the composite image. The embedded images retain easily recognizable when extracted. The embedded images can be extracted fairly completely even when lossy compression or cropping is applied to the composite image. The proposed method does not require the original cover image to extract the embedded image.

Keywords: image hiding, wavelet transform, watermark, compression, multiresolution

1. INTRODUCTION

Image hiding involves embedding images *E* into a cover image *C* to establish a composite image, *P*, that is perceptually similar to *C*. Image hiding is similar to digitally watermarking an image [1-3]. The difference is that image hiding conveys information about *E*, while a watermark conveys information about *C*. In the image hiding application, much more hidden information should be carried.

Image hiding methods can be divided into either spatial-domain or frequency-domain. Spatial-domain approaches are discussed in [4-9]. In these approaches, the hidden information is stored in the least significant bits (LSBs) of the pixels of the cover image. Spatial-domain techniques are intuitive but not robust. The composite image usually cannot be processed using operations such as intensity enhancement, resampling, requantization, image enhancement, cropping, and lossy image compression like JPEG.

Frequency domain techniques [10-20] take advantage of the human visual system's low sensitivity to high and middle frequency information. A common transform framework is the block-based discrete cosine transform (DCT) [10-20]. Typically, DCT-based methods divide the cover image into 8×8 pixel blocks and apply the DCT transform to each block. Note that hidden information inserted into the high frequencies is vulnerable to attack. Conversely, information insertion into the low frequencies may be seen. Thus,

Received August 4, 2003; revised December 17, 2004; accepted February 24, 2005.
Communicated by Pao-Choo Chung.
* This work was supported in part by the National Science Council of Taiwan, R.O.C., under grant No. NSC-90-2213-E-009-119.

in DCT approaches the middle-frequency information of the cover image is modified according to the embedded image to create the composite image. The embedded image can be extracted by subtracting the discrete cosine transforms of the composite image from the original image.

DCT based techniques require an original cover image *C* to extract *E* from *P*. Actually, many of the previous methods need *C* to extract *E* from *P* [1-14], but it is desirable to be able to extract *E* without knowing *C*. The ability to detect a hidden image without the original image introduces a very challenging problem especially if robustness is also desirable.

In this paper we present a wavelet-based image hiding method. The proposed method has several advantages. We can hide up to three full size images while maintaining the perceptibility of the extracted embedded images. The PSNR define value of the composite image is above 32dB, so that means the composite image is perceptually very similar to the original cover image, while the embedded image is not visible in the composite image. The proposed method is robust. Even when a quarter of the composite image is removed, we still can extract a perceptible embedded image. Furthermore, the embedded image is difficult to remove or steal. Images can be removed only when the high-pass filter and the low-pass filter employed in the wavelet transform are available. Finally, the proposed method does not require the original cover image to extract the embedded image.

In the next section, we will briefly review the wavelet transform. The proposed method is presented in section 3. Section 4 shows experimental results. We give conclusions in section 5.

2. 2-D DISCRETE WAVELET TRANSFORM

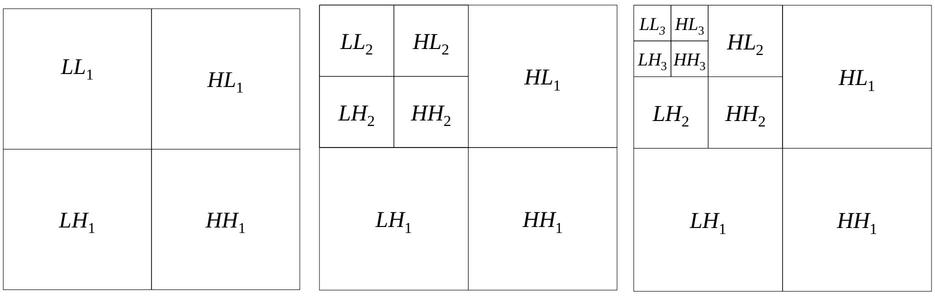
A two-dimensional discrete wavelet transform [21, 22] and its inverse are the extension of the one-dimensional discrete wavelet transform. They are implemented using a one-dimensional DWT and IDWT along each of the *x* and *y* coordinates. In other words, we apply a low-pass filter and a high-pass filter along each of the two coordinates. We then decompose the original image into four sub-images as follows:

1. *LL*: obtained by applying low-pass filters on both coordinates.
2. *HL* and *LH*: obtained by applying the high-pass filter on one coordinate and the low-pass filter on the other coordinate.
3. *HH*: obtained by applying the high-pass filter on both coordinates.

A commonly used wavelet transform is a three-stage wavelet transform (Eq. (1)) as shown in Fig. 1.

$$W(x, y, t) = DWT^3(f(x, y, t)), \text{ for } x, y\ \forall\ 0 \le t \le T - 1. \tag{1}$$

In the first stage of the transform, we split the original image into four quarter-size images, the upper left (LL_1), the upper right (HL_1), the lower left (LH_1), and the lower



(a) (b) (c)

Fig. 1. (a) The first stage of a 2-D wavelet transform, (b) a two-stage wavelet transform, and (c) a three-stage wavelet transform.

right (HH_1). In the subsequent stages, LL_i, $i \geq 1$, is recursively decomposed into four quarter-size components denoted LL_{i+1}, LH_{i+1}, HL_{i+1}, and HH_{i+1}.

Given the wavelet coefficients of an image $f(x, y)$, $f(x, y)$ can be reconstructed using the wavelet coefficients as well as LL_3. Note that, LL_3 is the region that contains the lowest frequency information. Because the human vision is sensitive to low spatial frequencies, LL_3 is the most important component in the reconstruction process.

3. THE PROPOSED METHODS

This section presents the methods for embedding one or three images into a cover image. We then show that recognizable embedded images can be extracted even when a part of the composite image is removed. This method was developed based on the following observations.

1. An image embedded in the high frequencies is vulnerable to attack. However, images embedded in the low frequencies may be visible.
2. LL_1, LL_2, and LL_3 in a three-stage wavelet transform are the most important components for reconstructing an embedded image. As long as these three components are available, we can reconstruct an image that is perceptibly similar to the original embedded image.

In the following, we use $C(x, y)$ to denote a cover image and $E(x, y)$ to denote an image to be hidden into $C(x, y)$.

3.1 Embedding One Image

Given two images $C(x, y)$ and $E(x, y)$, we establish an image $P(x, y)$ that is perceptually similar to $C(x, y)$. Furthermore, $P(x, y)$ carries an image $\hat{E}(x, y)$ that is perceptually similar to $E(x, y)$. $P(x, y)$ is established in the following way. We first expand $C(x, y)$ and $E(x, y)$ in terms of the three-stage wavelet coefficients as shown in Fig. 2. The wavelets coefficients are denoted X_{Y_i} where X represents an image, $Y = \{LL, HL, LH, HH\}$, and $i =$

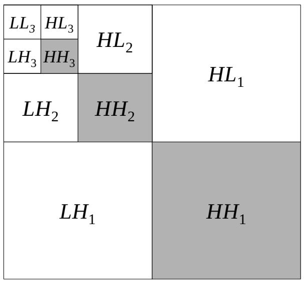


Fig. 2. The block HH_1 is replaced by E_{LL_1}, HH_2 is replaced by E_{LL_2}, and HH_3 is replaced by E_{LL_3}.

1, 2, 3 for the three scales. To construct the image $\hat{E}(x, y)$ with the least amount of information possible, we use the low frequency components E_{LL_1}, E_{LL_2}, and E_{LL_3}. We embed E_{LL_1}, E_{LL_2}, and E_{LL_3} into $C(x, y)$ to establish $P(x, y)$. Because human vision is not very sensitive to high spatial frequencies, we replace the wavelet coefficients C_{HH_1} with E_{LL_1}, C_{HH_2} with E_{LL_2}, and C_{HH_3} with E_{LL_3} (Fig. 2). The inverse wavelet transform is then applied to the resulting image to establish $P(x, y)$.

Extraction $\hat{E}(x, y)$ starts with applying a three-stage wavelet transform to $P(x, y)$. Note that P_{HH_1}, P_{HH_2}, and P_{HH_3} are the components replaced by E_{LL_1}, E_{LL_2}, and E_{LL_3}. If we take P_{HH_1} as the low frequency component of an image $E1_{LL_1}$, and assign $E1_{HL_1}$, $E1_{LH_1}$, and $E1_{HH_1}$ as zero, we can reconstruct the image $E1$. Similarly, we can obtain $E2$ and $E3$ from P_{HH_2}, and P_{HH_3}. Because $E1$, $E2$, and $E3$ are reconstructed from the low frequency components of $E(x, y)$, these images are perceptually similar to $E(x, y)$. Finally, $\hat{E}(x, y)$ is then obtained by averaging the three images $E1$, $E2$, and $E3$.

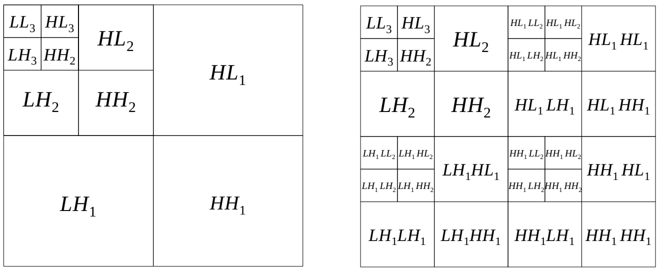
3.2 Embedding Tree Images

This wavelet approach can be extended to embed three images into a cover image. In the proposed extension, a composite wavelet transform is employed to decompose the cover image.

Given a cover image $C(x, y)$ and three images $E^1(x, y)$, $E^2(x, y)$, and $E^3(x, y)$ that are embedded into $C(x, y)$, we apply a three-stage wavelet transform to each $E^i(x, y)$. $C(x, y)$ is processed differently. We first apply a three-stage wavelet transform to $C(x, y)$, and then we apply a two-stage wavelet transform to each of HL_1, LH_1, and HH_1. The resultant composite wavelet transform is shown pictorially in Fig. 3 (b).

To embed three images into $C(x, y)$, we substitute certain components of $C(x, y)$ with those of $E^i(x, y)$ as shown in Fig. 4, and as follows:

C_{HH_3} is replaced by $E^1_{LL_3}$.
$C_{HL_1HH_2}$ is replaced by $E^1_{HL_3}$.
$C_{LL_1HH_2}$ is replaced by $E^1_{LH_3}$.
$C_{HH_1HH_1}$ is replaced by $E^1_{HH_2}$.
$C_{HL_1HH_1}$ is replaced by $E^2_{LL_2}$.
$C_{LH_1HH_1}$ is replaced by $E^3_{LL_2}$.



(a) (b)

Fig. 3. The wavelet transform applied to $C(x, y)$ in the case of three embedded images.

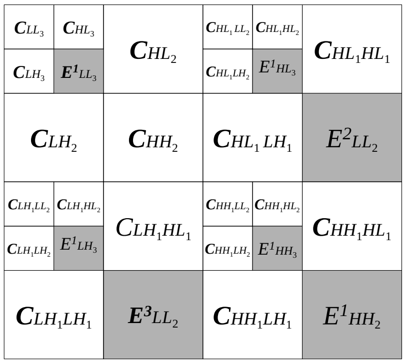


Fig. 4. The substitutions for embedding three images.

Appling an inverse of this resultant composite wavelet transform to the image after the substitutions stated above establishes the composite image $P(x, y)$.

Extracting the three embedded images starts with the composite wavelet transforms. The replaced components obtained from the three embedded images are employed to reconstruct $\hat{E}^1(x, y)$, $\hat{E}^2(x, y)$, and $\hat{E}^3(x, y)$.

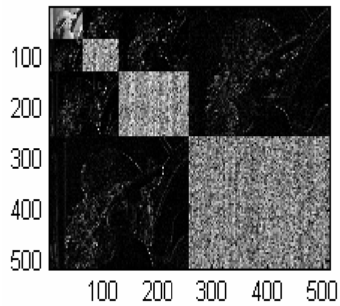
3.3 Scrambling the Embedded Information

Scrambling the embedded information involves rearranging the pixels in the substituted images (E_{LL_1}, E_{LL_2}, and E_{LL_3} for embedding one image, or $E^1_{LL_3}$, $E^1_{HL_3}$, $E^1_{LH_3}$, $E^1_{HH_2}$, $E^2_{LL_2}$, and $E^3_{LL_2}$ for embedding three images). Scrambling is to ensure security and robustness.

1. If a scrambler is applied, the embedded images can be extracted only when the scrambling function and the low-pass and high-pass filters are available.
2. Because the embedded information is scrambled, neighboring pixels in the embedded information are spread over the entire area of the composite image after IDWT. Suppose that an area in $P(x, y)$ is cut away, then part of the embedded information is also

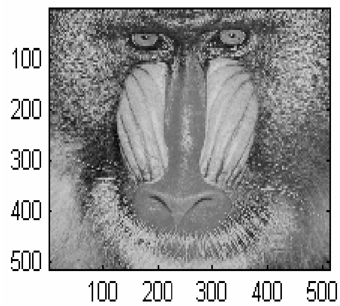
lost. Because the lost information is a set of non-neighboring pixels in the embedded image, we are still able to extract perceptible embedded image(s). The scrambling functions are implemented using a scrambling array which itself is generated using a random number generator.

(a) Composite image (PSNR = 32.51dB).



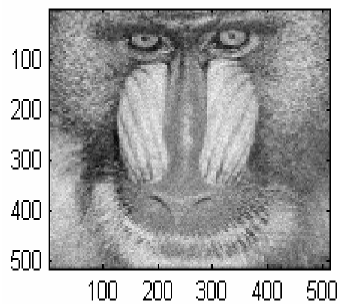
(b) Wavelet domain.

(c) Cover image.



(d) Embeded image 1.

(e) Extracted cover image (PSNR = 33.12dB).



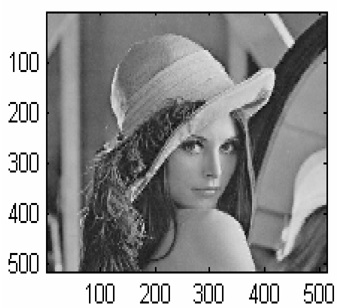
(f) Extracted image 1 (PSNR = 22.27dB).

Fig. 5. (a) Composite image. (b) Composite image in the wavelet domain. (c) Original cover image. (d) Original embedded image. (e) Extracted cover image. (f) Extracted embedded image.

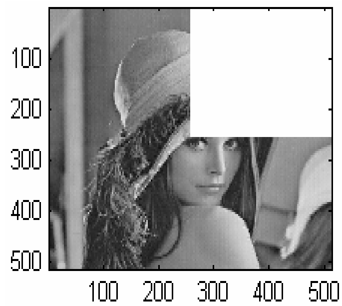
4. EXPERIMENTAL RESULTS

In this section, we present experimental results. In the case of embedding one image, we embed an image (Fig. 5 (d)) that has the same size (512 by 512) as the cover image (Fig. 5 (c)). Fig. 5 (a) shows the composite image. The PSNR value of the composite image is above 32.51 dB. Fig. 5 (b) shows the wavelet domain image in which the embedded information is scrambled. Fig. 5 (e) shows the cover image after the embedded image was extracted. It has PSNR 33.12 dB compared to the original cover image. Fig. 5 (f) shows the extracted embedded image. It has PSNR 22.27 dB compared to the original embedded image. Although the PSNR is less than 30 dB, Fig. 5 (f) is perceptually similar to Fig. 5 (d).

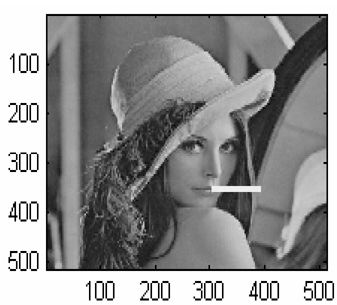
Fig. 6 demonstrates the resistance of the proposed method to cropping. In Fig. 6 (a), a small portion of the composite image was removed. Fig. 6 (c) shows the cover image after embedded images was extracted and Fig. 6 (e) is the extracted embedded image. The extracted embedded image has PSNR 20.50 dB. Even though it has a low PSNR, it can still be easily recognized. Fig. 6 (b) shows a case in which we cut a quarter of the composite image away. Fig. 6 (d) shows the cover image after the embedded image was extracted. The extracted embedded image is shown in Fig. 6 (f). The PSNR dropped to 13.56 dB. However, it is still recognizable.



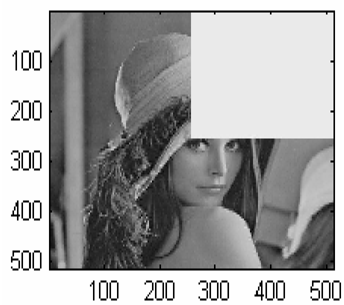
(a) Composite image (PSNR = 25.61dB).



(b) Composite image (PSNR = 11.84dB).

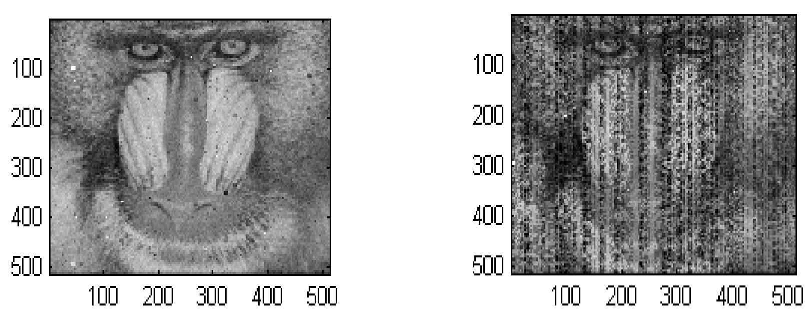


(c) Extracted cover image (PSNR = 25.57dB).



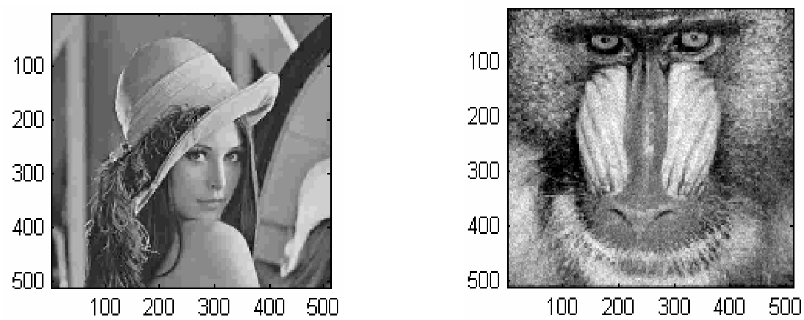
(d) Extracted cover image (PSNR = 11.90dB).

Fig. 6. (a) Composite image containing a small missing part. (b) Cover image after the embedded image was extracted. (c) Extracted embedded image. (d) Upper right quarter of the composite image was removed. (e) Cover image after the embedded image was extracted. (f) Extracted embedded image.

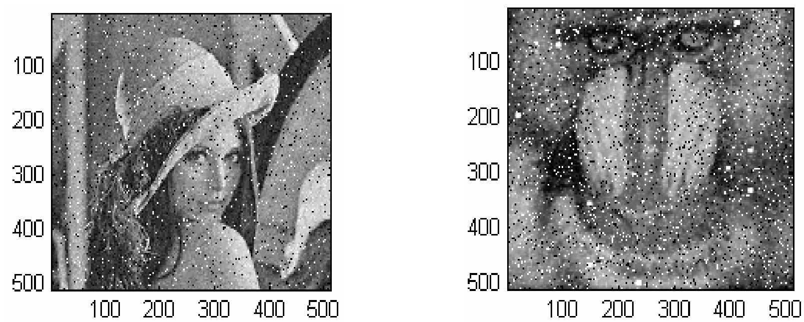


(e) Extracted image (PSNR = 20.50dB). (f) Extracted image 1 (PSNR = 13.56dB).

Fig. 6. (Cont'd) (a) Composite image containing a small missing part. (b) Cover image after the embedded image was extracted. (c) Extracted embedded image. (d) Upper right quarter of the composite image was removed. (e) Cover image after the embedded image was extracted. (f) Extracted embedded image.



(a) Composite image+JPEG (PSNR = 21.36dB). (b) Extracted image from (a) (PSNR =18.21dB).



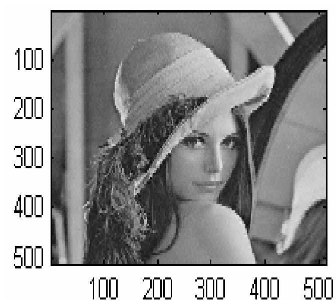
(c) Composite image+Nosie+JPEG (PSNR = 18.74dB). (d) Extracted image from (c) (PSNR = 16.83dB).

Fig. 7. (a) Composite image after JPEG compression. (b) Extracted embedded image from (a). (c) Composite image after JPEG compression and adding noise. (d) Extracted embedded image from (c).

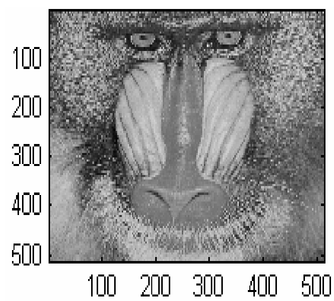
Fig. 7 (a) shows the composite image after being compressed using JPEG with a 90% compression ratio. The extracted embedded image is shown in Fig. 7 (b). Fig. 7 (c) shows the compressed composite image with noises added. The extracted embedded image is shown in Fig. 7 (d). These results show that, after JPEG compression and adding noise, a perceptible embedded image can still be extracted.

This method can embed up to three images while maintaining recognizability of the three extracted embedded images. In this experiment, the cover image was the same as in the previous experiment. Fig. 8 (a) shows the composite image, which has PSNR 32.16 dB and Fig. 8 (c), (e) and (g) are the three embedded images. All images are 512 by 512 pixels. Fig. 8 (b) shows the extracted cover image, which has PSNR 32.91 dB. Fig. 8 (d), (f) and (h) show the extracted images, which have PSNR values of 17.65 dB, 22.48 dB, and 4.59 dB, respectively.

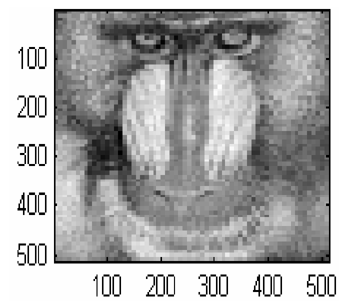
(a) Composite image (PSNR = 32.16dB).



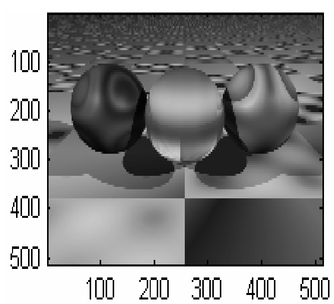
(b) Extracted cover image (PSNR= 32.91dB).



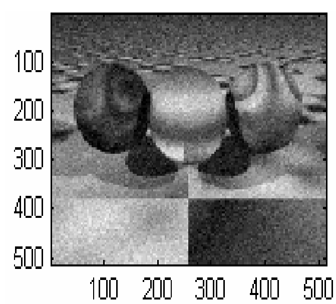
(c) Embeded image 1.



(d) Extracted image 1 (PSNR = 17.65dB).

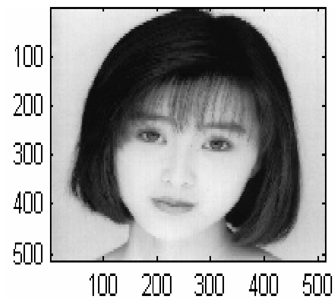


(e) Embeded image 2.

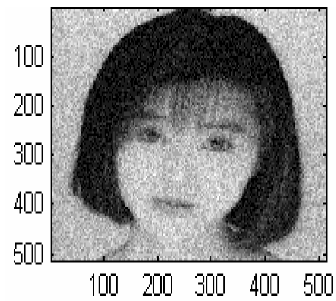


(f) Extracted image 2 (PSNR = 22.46dB).

Fig. 8. (a) Composite image. (c, e, g) The three embedded images. (b) Extracted cover image. (d, f, h) Extracted embedded images.



(g) Embeded image 3.



(h) Extracted image 3 (PSNR = 4.59dB).

Fig. 8. (Cont'd) (a) Composite image. (c, e, g) The three embedded images. (b) Extracted cover image. (d, f, h) Extracted embedded images.

5. CONCLUSIONS

We presented an easy, yet effective method to embed one to three images into another image. The PSNR of the composite images were above 32 dB for cases when one image or three images were embedded. In both cases, perceptible embedded images can be extracted. This method is very robust. We have shown that we can extract recognizable embedded image even when one quarter of the composite image was removed. When other operations, such as JPEG compression and adding noise, were applied to the composite images, the embedded images can still be extracted.

In all of our experiments, we embedded one full size image or three full size images. If the size of the embedded image is smaller, the PSNR of the extracted embedded images would be further improved. We believe that the robustness can also be improved.

REFERENCES

1. E. Koch and J. Zhao, "Toward robust and hidden image copyright labeling," in *Proceedings of the IEEE Nonlinear Signal and Image Processing*, 1995, pp. 452-455.
2. I. J. Cox, J. Kilian, T. Leighton, and T. Shammoon, "Secure spread spectrum watermarking for multimedia," Technical Report 95-10, NEC Res. Inst., Princeton, NJ, 1995.
3. M. D. Swanson, B. Zhu, and A. H. Tewfik, "Transparent robust image watermarking," in *Proceedings of the IEEE International Conference on Image Processing*, 1996, pp. 211-214.
4. L. F. Turner, "Digital data security system," *Patent IPN WO 89/08915*, 1989.
5. S. Walton, "Image authentication for a slippery new age," *Dr. Dobb's Journal*, Vol. 20, 1995, pp. 18-26.
6. D. C. Wu and W. H. Tsai, "Image hiding in spatial domain using an image differencing approach," in *Proceedings of Conference on Computer Vision, Graphics, and Image Processing*, 1998, pp. 280-287.
7. I. Pitas, "A method for signature casting on digital images," in *Proceedings of the*

IEEE International Conference on Image Processing, Vol. 341, 1996, pp. 215-218.
8. W. Bender, D. Gruhl, and N. Morimoto, "Techniques for data hiding," in *Proceedings of the SPIE Conference on Storage and Retrieval for Image and Video Databases III*, Vol. 2420, 1995, pp. 164-173.
9. O. Bruyndonckx, J. J. Quisquater, and B. Macq, "Spatial method for copyright labeling of digital images," in *Proceedings of the IEEE Workshop on Nonlinear Signal and Image Processing*, 1995, pp. 456-459.
10. A. Piva, M. Barni, F. Bartolini, and V. Cappellini, "DCT-based watermark recovering without resorting to the uncorrupted original image," in *Proceedings of the IEEE International Conference on Image Processing*, Vol. 1, 1997, pp. 520-523.
11. C. T. Hsu and J. L. Wu, "Hidden digital watermarks in images," *IEEE Transactions on Image Processing*, Vol. 8, 1999, pp. 58-68.
12. H. Kii, J. Onishi, and S. Ozawa, "The digital watermarking method by using both patchwork and DCT," in *Proceedings of the IEEE Multimedia Computing and System*, Vol. 1, 1999, pp. 895-899.
13. M. Wu and B. Liu, "Watermarking method for image authentication," in *Proceeding of the IEEE International Conference on Image Processing*, Vol. 2, 1998, pp. 437-411.
14. W. Zeng and B. Liu, "On resolving rightful ownerships of digital images by invisible watermarks," in *Proceeding of the IEEE International Conference on Image Processing*, Vol. 1, 1997, pp. 552-555.
15. C. C. Chang, J. C. Yeh, and J. Y. Hsiao, "A method for protecting digital images from being copied illegally," in *Proceedings of the International Conference on Digital Libraries: Research and Practice*, 2000, pp. 373-379.
16. F. Baldo, F. Perez-Gonzalez, and S. Scalise, "Turbo coding for sample-level watermarking in the DCT domain 2001," in *Proceedings of the IEEE International Conference on Image Processing*, Vol. 3, 2001, pp. 1003-1006.
17. M. S. Hsieh, D. C. Tseng, and Y. H. Huang, "Hiding digital watermarks using multiresolution wavelet transform," *IEEE Transactions on Industrial Electronics*, Vol. 48, 2001, pp. 875-882.
18. M. Y. Huang, D. C. Tseng, and M. S. C. Liu, "Wavelet image enhancement based on Teager energy operator," in *Proceedings of the International Conference on Pattern Recognition*, 2002, pp. 993-996.
19. G. Xuan, J. Zhu, J. Chen, Y. Q. Shi, Z. Ni, and W. Su, "Distortionless data hiding based on integer wavelet transform," *IEEE Electronics Letters*, Vol. 38, 2002, pp. 1646-1648.
20. C. T. Hsu and J. L. Wu, "Multiresolution watermarking for digital images," *IEEE Transactions on Circuits and Systems II*: *Analog and Digital Signal Processing*, Vol. 45, 1998, pp. 1097-1101.
21. S. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence* (*PAMI*), Vol. 11, 1999, pp. 647-693.
22. P. P. Vaidyanathan, *Multirate Systems and Filter Banks*, Prentice Hall, Inc., 1993.

Chih-Yang Lin (林志陽) received the Master degree in Computer Science and Engineering from Yuan Ze University, Taiwan, in 1996. He was an assistant manager at R&D department of Chilong Co. Ltd. from 1988 to 1993. He was a senior engineer at Motorola Electronics Ltd. in Taiwan from 1993 to 1996. Currently he is a Ph.D. candidate in Computer and Information Science at the National Chiao Tung University. His current research interests include medical imaging, signal process, and wavelets.

Yu-Tai Ching (荊宇泰) received the B.S. degree in Industrial Engineering from Tsing Hua University, Taiwan, R.O.C., in 1980, and the M.S. and Ph.D. degrees in Computer Science from Northwestern University, Evanston, IL, in 1983 and 1987. He is currently an Associate Professor in the Department of Computer and Information Science at National Chiao Tung University, Hsinchu, Taiwan. His research interests are design and analysis of computer algorithms, medical image analysis, and computer graphics.