JPEG2000-BASED DATA HIDING TO SYNCHRONOUSLY UNIFY DISPARATE FACIAL DATA FOR SCALABLE 3D VISUALIZATION

K. HAYAT, W. PUECH and G. SUBSOL

LIRMM, UMR CNRS 5506, University of Montpellier II,
161, rue Ada, 34392 Montpellier CEDEX 05, France
E-mail: khizar.hayat@lirmm.fr, william.puech@lirmm.fr and gerard.subsol@lirmm.fr
www.lirmm.fr

G. GESQUIERE

LSIS, UMR CNRS 6168, Aix-Marseille University
IUT, rue R. Follereau, 13200 Arles, France
E-mail: gilles.gesquiere@lsis.org
www.lsis.org

We present a scalable encoding strategy for the 3D facial data in various bandwidth scenarios. The scalability, needed to cater diverse clients, is achieved through the multiresolution characteristic of JPEG2000. The disparate 3D facial data is synchronously unified by the application of data hiding wherein the 2.5D facial model is embedded in the corresponding 2D texture in the discrete wavelet transform (DWT) domain. The unified file conforms to the JPEG2000 standard and thus no novel format is introduced. The method is effective and has the potential to be applied in videosurveillance and videoconference applications.

Keywords: JPEG2000, discrete wavelet transform (DWT), scalability, synchronization, videosurveillance, videoconference, data hiding, LSB, 3D, 2.5D, face data, multiresolution

1. Introduction

Transmitting digital 3D face data in real-time has been a research issue for quite a long time. When it comes to the real-time, two main areas, viz. conferencing and surveillance, suddenly come to mind. In the earlier videoconference applications, the aim was to change the viewpoint of the speaker. This allowed in particular to recreate a simulation replica of a real meeting room by visualizing the "virtual heads" around a table.[1] Despite

the fact that many technological barriers have been eliminated, thanks to the availability of cheap cameras, powerful graphic cards and high bitrate networks, there is still no commercial product that offers a true conferencing environment. Some companies, such as Tixeo in France[a], propose a 3D environment where interlocutors can interact by moving an avatar or by presenting documents in a perspective manner. Nevertheless, the characters remain artificial and do not represent the interlocutors' real faces. In fact, it seems that changing the viewpoint of the interlocutor is considered more as a gimmick than a useful functionality. This may be true of a videoconference between two people but in the case of a conference that would involve several interlocutors spread over several sites that have many documents, it becomes indispensable to replicate the conferencing environment. Another application consists in tracking the 3D movement of the face in order to animate a clone, i.e. a model of the users face. In fact, the transmission of only a small number of parameters of movement or expression can materialize the video through low speed networks. However, recent technology have increased the bandwidth of conventional telephone lines to several Mbps. This has led to a slowing down of research activities on the subject in recent years. Nevertheless, the bitrate limitation still exists in the case of many devices like PDA or mobile phones. It becomes even critical, in particular in remote videosurveillance applications which are gaining increasing economic importance. Some companies offer to send surveillance images on the mobile phones/PDAs of authorized persons but these are only 2D images whereby the identification of persons is very difficult, especially in poor light conditions.

Arguably the most important factor in 3D face transmission is the network bandwidth. Alongside is the diversity of clients in terms of the computing and memory resources, distance, network and need. All these factors compel us to think of strategies to not only compress the data, without compromising on the visual quality, but also have some scalability to cater for the diversity of clients. The objective of this work is to reduce the data considerably for optimal real-time 3D facial visualization in a client/server environment. 3D face data essentially consists of a 2D color image called texture and its corresponding depth map in the form of what is called 2.5D image. The latter is usually obtained[2] by the projection of the 3D polygonal mesh model onto the image plane after its normalization. For 3D visualization one would thus have to manipulate at least two files. It would be

---

[a]www.tixeo.com

better to have a single file rather than two. For this purpose we propose to unify the two files into a single standard JPEG2000 format file. The use of DWT-based JPEG2000 will give us two main advantages. One, the multiresolution nature of wavelets would offer the required scalability to make for the client diversity. Two, we will not be introducing any new file format but conform to a widely known standard. To ensure highest quality for a resource rich client we would use the JPEG2000 codec in the lossless mode. For the unification of the 2D texture and 2.5D model, a scalable data hiding strategy is proposed wherein the 2.5D data is embedded in the corresponding 2D texture in the wavelet transform domain. This would allow to transmit all the data in a hierarchical and synchronized manner. The idea is to break down the image and its 3D model at different levels of resolution. Each level of resolution of the image will contain the associated 3D model without reducing the image quality and without any considerable increase the file size.

The rest of this paper is arranged as follows. Section 2 gives a concise survey of the data hiding methods in the wavelet domain. The proposed method is explained in Section 3 and the results obtained by its application to a number of examples from FRAV3D[b] database are summarized in Section 4. Concluding remarks, with special reference to the future work, are given in Section 5.

2. State of the art

The essence of data hiding is the embedding of an information, called message, inside some host signal, like image, sound or video, called cover. The message may be small with robust embedding requirement as in the case of copyright protection in the form of watermarking or it may be large, critical and statistically invisible as in steganography. Four factors[3] characterize the effectiveness of a data hiding method, namely embedding capacity, perceptual transparency, robustness and tamper resistance. Embedding capacity refers to the maximum payload that can be held by the cover. Perceptual transparency ensures the retention of visual quality of the cover after data embedding. Robustness is the ability of the cover to withstand various signal operations, transformations and noise whereas tamper resistance means to remain intact in the face of malicious attacks. The relative importance of these four factors depends on the particular data hiding application. For example, for visually sensitive applications perceptual transparency becomes

---

[b]www.frav.es/databases/FRAV3d/

very important. Domain-wise, embedding can be carried out in the spatial domain or the transform domain. Pixel or coefficient allocation for data embedding may be regular (*e.g.* every k^{th} pixel) or irregularly distributed (*e.g.* pseudo-random). Probably the most preferred pixel allocation is by running a pseudo-random number generator (PRNG) using some secret key as a seed. Finally, an embedding method is *blind* if data extraction by the recipient does not require the original cover.

Data hiding methods for JPEG2000 images must process the code blocks independently.[4] That is why a majority of the wavelet-based data hiding methods proposed in the literature not compatible with the JPEG2000 scheme. There are methods[5,6] for embedding invisible watermarks by adding pseudo-random codes to large coefficients of the high and middle frequency bands of DWT but the methods have the disadvantage of being non-blind. Piva *et al.* have proposed an authentication scheme that embeds an image digest in a subset of the subbands from the DWT domain.[7] The image digest is derived from the DCT of the level 1 DWT LL subband of the image. The resultant DCT coefficients are scaled down by quantization and ordered from most to least significant through a zig-zag scan.

One blind method[8] transforms the original image by one-level wavelet transform and sets the three higher subbands to zero before inverse transforming it to get the modified image. The difference values between the original image and the modified image are used to ascertain the potential embedding locations of which a subset is selected pseudo-randomly for embedding. For the sake of robustness and perceptual transparency Kong *et al.*[9] embeds watermark in the weighted mean of the wavelets blocks, rather than in the individual coefficient. While explaining their method of embedding biometric data in fingerprint images, Noore *et al.* argue against the modification of the lowest subband to avoid degradation of the reconstructed image as most of the energy is concentrated in this band.[10] Instead they propose to redundantly embed information in all the higher frequency subbands.

Uccheddu *et al.*[11] adopt a wavelet framework in their blind watermarking scheme for 3D models under the assumption that the host meshes are semi-regular ones paving the way for a wavelet decomposition and embedding of the watermark at a suitable resolution level. For the sake of robustness the host mesh is normalized by a Principal Component Analysis (PCA) before embedding. Watermark detection is accomplished by computing the correlation between the watermark signal and the "to-be-inspected" mesh. Yu *et al.*[12]propose a robust 3D graphical model watermarking scheme for

triangle meshes that embeds watermark information by perturbing the distance between the vertices of the model to the center of the model. With robustness and perceptual transparency in focus, the approach distributes information corresponding to a bit of the watermark over the entire model. The strength of the embedded watermark signal is adaptive with respect to the local geometry of the model. Yin *et al.*[13] adopt Guskov's multiresolution signal processing method for meshes and use his 3D non-uniform relaxation operator to construct a Burt-Adelson pyramid for the mesh, and then watermark information is embedded into a suitable coarser mesh. The algorithm is integrable with the multiresolution mesh processing toolbox and watermark detection requires registration and re-sampling to bring the attacked mesh model back into its original location, orientation, scale, topology and resolution level.

3. The Proposed Method

For a $N \times N$ pixel facial texture and its corresponding $M \times M$ point depth map $(2.5D)$ we propose our data hiding strategy presented in Fig. 1. The face texture is subjected to the level-L JPEG2000 encoding in the lossless mode. The encoding process is interrupted after the DWT step to get the three transformed YCrCb face texture components. The corresponding grayscale $(k-1$ bit) depth map is also subjected to level-L lossless DWT in parallel. To ensure the accuracy we expand the word-size for each of the transformed depth map coefficient by one additional bit and represent it in k bits. The DWT domain depth map coefficients are then embedded in the DWT domain YCrCb face texture components while strictly following the spatial correspondence, i.e. low frequency $2.5D$ coefficients in low while higher in higher frequency YCrCb coefficients. This step strictly depends on the ratio, $M : N$, where $M \leq N$. In the worst case, where $M = N$, the k bit transformed $2.5D$ coefficient is equally distributed among the three components and each of the transformed YCrCb texture coefficient carry $\lfloor k \rfloor$ to $\lfloor k \rfloor + 1$ bits. If $M < N$ then, rather than a face texture coefficient, a whole face texture block corresponds to one depth map coefficient and one has the choice of selecting the potential carrier coefficients. This is specially true when $M < N/3$ as one is then compelled to run a pseudo-random generator (PRNG) to select the potential carrier coefficients. To keep the method blind, the embedding process involves the substitution of the least significant bit (LSBs) of the carrier coefficient with the bit(s) from the 2.5D coefficient. After embedding, the YCrCb component are re-inserted into the JPEG2000 coding pipeline. The result is a monolithic JPEG2000
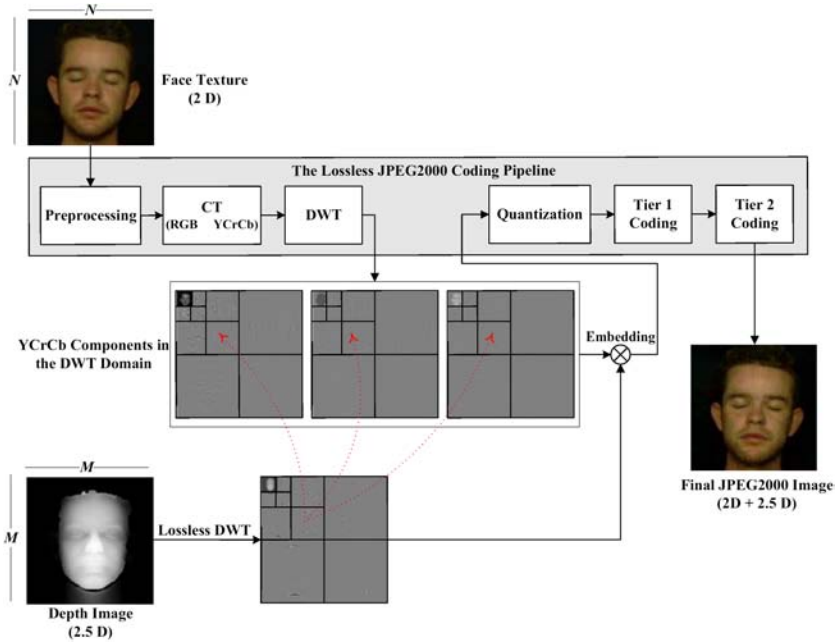
Fig. 1: Description of the method

format face texture image that has the depth map hidden in it.

A precursor of this work can be found in Ref. 14 wherein the method was developed for 3D terrain visualization. In that scenario we had the luxury of choosing the potential carrier coefficients from a large population of texture coefficient since M was very small as compared to N. For the work in perspective we have chosen the worst case scenario, i.e. $M = N$. This would also help us to have an idea of the embedding capacity for the earlier work. In the embedding step, a given k-bit transformed depth map coefficient is to be substituted into the $[k/3]$ LSBs each of the corresponding Y, Cr and Cb transformed coefficients. To reduce the payload we have optimized our method to some extent. One of the important characteristics of DWT is the high probability of 0 coefficients in higher frequency subbands. Hence one can always use a flag bit to differentiate this case from the rest. In addition, the use of kth additional bit for transform domain coefficients is a bit too much. Thus, for example, for an 8 bit spatial domain $2.5D$ coefficient the initial range of $[-128, 127]$ may not be enough in the DWT domain and needs to be enhanced but not to the extent to warrant a range of $[-256, 255]$.

A midway range of $[-192, 192]$ ought to be sufficient. For such a 8-bit scenario one may then have four possibilities for the value of a coefficient viz. zero, normal ($[-128, 127]$), extreme negative ($[-192, -128]$) and extreme positive ($[128, 192]$). Keeping all these possibilities in view, we decided to pre-process the transformed depth coefficient set, before embedding. In our strategy, we keep the first bit exclusively as a flag bit. The next two bits are data cum flag bits and the last six bits are strictly data bits. For a coefficient in the range $[-128, 127]$, the first bit is set to 0, with the rest of eight bits carrying the value of the coefficient, otherwise it is set to 1. For a zero coefficient, the first two bits are set to 1 and thus only 11 is inserted. The absolute difference of an extreme negative coefficient and -128 is carried by the last six bits with the first three bits carrying 101. For extreme positives the first three bits have 100 and the rest of six bits have the absolute difference of the coefficient with $+127$. In essence we are to embed either two or nine bits according to the following policy:

- **if** $coeff \in [-128, 127]$ then concatenate coeff to 0 and embed as 9 bits;
- **else if** $coeff = 0$ then embed binary 11;
- **else if** $coeff \in [-192, -128]$ then concatenate $|-128 - coeff|$ to 101 and embed as 9 bits;
- **else** concatenate ($coeff - 128$) to 100 and embed as 9 bits;

The above coded image can be utilized like any other JPEG2000 image and sent across any communication channel. The blind decoding is the reverse of the above process.

The method thus enables to effect visualization from a fraction of data in the form of the lowest subband, of a particular resolution level since it is always possible to stuff 0's for the higher bands. The idea is to have a $3D$ visualization utilizing lower frequency subbands at level L', with $L' \leq L$. For the rest of $L - L'$ parts one can always pad a 0 for each of their coefficient. The inverse DWT of the 0-stuffed transform components will yield what is known as image of approximation of level L'. A level-L' approximate image is the one that is constructed with $(1/4^{L'}) \times 100$ percent of the total coefficients that corresponds to the available lower $3L' + 1$ subbands. For example, level-0 approximate image is constructed from all the coefficients and level-2 approximate image is constructed from 6.12% of the count of the initial coefficients. Before being subjected to inverse DWT, data related to depth map must be extracted from the transformed face texture whose size depends both L and L'. Thus if $L' = L$ one will always have the entire

set of the embedded DEM coefficients since all of them will be extractable. We would have a level 0 approximate final DEM after inverse DWT, of the highest possible quality. On the other hand if $L' < L$, one would have to pad 0's for all coefficients of higher $L - L'$ subbands of transformed DEM before inverse DWT that would result in a level L'-approximate DEM of an inferior quality.

4. Experimental Results

We have applied our method to a number of examples from FRAV3D[c] database. One such example is given in Fig. 2 that consists of a 120×120 point 2.5D depth map (Fig. 2.a) corresponding to a 120×120 pixel colored 2D face image given in Fig. 2.b. Each point of the 2.5D depth map is coded with 8 bits. The 3D visualization based on the two images is depicted by a view given in Fig. 2.c. Lossless DWT is applied in isolation to the depth map at level-3 to get the image given in Fig. 3.a. To ensure the accuracy we represent each of the transformed depth map in 9 bits. The corresponding 2D face image is subjected to level-3 lossless JPEG2000 encoding and the process is interrupted just after the DWT step. What we get are the level-3 transformed luminance and chrominance components given in Fig. 3b-d. The transformed depth map is embedded in the three components according to the scheme outlined above. The resultant components are reintroduced to the JPEG2000 pipeline at quantization step. The final result is a single JPEG2000 format 2D image.

| (a) | (b) | (c) |

Fig. 2: Original data: a) 120×120 depth map (2.5D), b) The corresponding 120×120 2D face image, c) A 3D face view obtained from (a) and (b)

As already stated, level-L approximate image is the one that is constructed with $(1/4^L) \times 100$ percent of the total coefficients that corresponds

---

[c]www.frav.es/databases/FRAV3d/

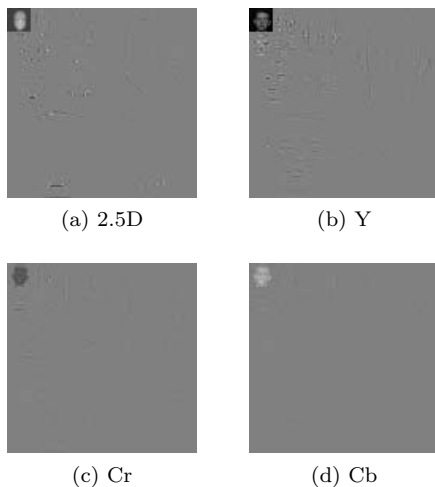(a) 2.5D   (b) Y

(c) Cr   (d) Cb

Fig. 3: Level-3 DWT domain images: a) Depth map after the application of lossless DWT, b-d) Components of the transformed 2D face image from the lossless JPEG2000 coding pipeline.

to the available lowest frequency $3L+1$ subbands. The level-3 encoded image with our method can give us four different quality 2D/2.5D pairs upon decoding and reconstruction. In terms of increasing quality, these are level-3, 2, 1 and 0 images reconstructed from 1.62%, 6.25%, 25% and 100% of the transmitted coefficients, respectively. The number of lowest subbands involved being $1, 4, 7$ and 10 out of the total of 10 subbands, respectively. For visual comparison, the approximation 2D images are given in Fig 4 while the approximation depth maps are shown in Fig 5.

Approximation image	lev. 3	lev. 2	lev. 1.	lev. 0
Bitrate (*bpp*)	0.41	1.10	3.55	8.38
MSE	120.25	43.35	21.17	20.16
PSNR (*dB*)	27.33	31.76	34.87	35.09

Table 1: Average results obtained for 2D face images after the extraction and reconstruction as a function of the transmitted data (*for level 0, all of the transmitted coefficients are used for reconstruction*).

For the purpose of quantitative comparison the average results over all the FRAV3D $2D/2.5D$ pairs subjected to our method are tabulated in
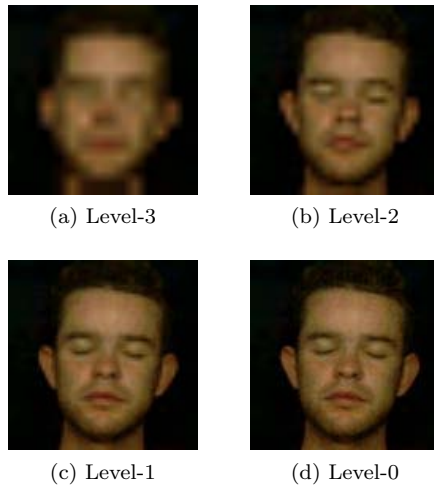
(a) Level-3   (b) Level-2

(c) Level-1   (d) Level-0

Fig. 4: Approximation 2D images obtained after the decoding and reconstruction .

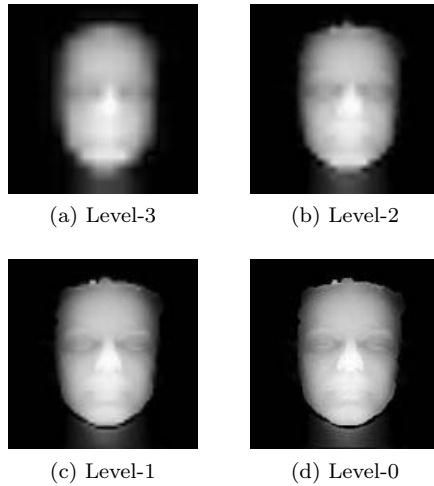(a) Level-3   (b) Level-2

(c) Level-1   (d) Level-0

Fig. 5: Approximation 2.5D images obtained after the decoding and reconstruction from the embedded image.

Table 1 and Table 2. For the 2D face images it can be observed that the level-3 approximate image is the lowest quality having a mean PSNR of 27.33 dB which is not bad in the face of the fact that it is constructed from just 0.25%

Approximation image	lev. 3	lev. 2	lev. 1.	lev. 0
Bits per coefficient (theoretical)	0.14	0.56	2.25	9
Bits per coefficient (optimized)	0.14	0.44	1.49	4.98
RMSE	11.33	7.76	4.73	0
PSNR (dB)	27.05	30.33	34.63	∞

Table 2: Average results obtained for the depth maps after the extraction and reconstruction as a function of the transmitted data (*for level 0, all of the transmitted coefficients is used for reconstruction*).

(a) Level-3

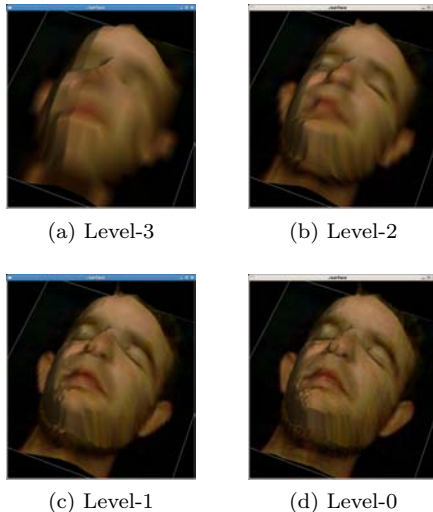(b) Level-2

(c) Level-1

(d) Level-0

Fig. 6: A 3D view from the visualization with the 2D/2.5D Approximation pairs at different levels.

of the transmitted coefficients. The level-0 approximate face image has a mean PSNR of 35.09 dB despite the fact that we are treating the worst case, i.e. both the 2D and 2.5D have the same dimensions. Even doubling the 2D dimensions, i.e. one depth map point corresponds to four 2D pixels, gave us a PSNR of 49.05 dB. For 2.5D approximations we are comparing the theoretical or worst case compression to that obtained by the application of our method in Table 2. It can be seen that for very high frequency the probability of zero is high and that is why for level-0 approximation we observed a mean bitrate of 4.98 against the expected value of 9. Since level-3 approximation has only the lowest frequency subband, the bitrate stays

at 0.14 for both. We have used root mean square error (RMSE) as an error measure in length units for $2.5D$. The 3D visualization obtained from the approximation $2D/2.5D$ pairs are depicted in the form of a 3D view at a particular angle in Fig. 6.

5. Conclusion

The results have been interesting in the sense that even with a depth map of the same dimensions as the 2D face image one got a good quality visualization. Usually the sizes are not the same and one depth map coefficient corresponds to a square block of 2D face texture pixels. Even for a 2×2 block the PSNR for level-0 jumps in average from 35.09 dB to 49.05 dB. The trend in our results shows that an effective visualization is possible even with a 0.1% of the transmitted coefficients, i.e. level-5. This must bode well for our ultimate goal of videoconferencing and videosurveillance when frames would replace the still image. Hence for a client with meager computing, memory or network resources a tiny fraction of the transmitted data should do the trick. The scalability aspect can then hierarchically take care of resourceful clients. This video aspect is the focus of our future work but before that one would like to further preprocess the depth map data before embedding. We had applied a raw optimization but this aspect must be further refined in the face of the fact that zeros and near-zeros in low frequency subbands are usually replicated in higher frequency subbands. Thus a kind of probability distribution is imminent which needs to be investigated. In the near future we may well apply the concept of embedded zero-tree wavelets (EZW[15]) to the data before embedding. For critical face features the application of region of interest coding (ROI) must also be investigated.

Acknowledgment

References

1. Weik, S., Wingbermuhle, J., and Niem, W., "Automatic Creation of Flexible Antropomorphic Models for 3D Videoconferencing," in [*Proceedings of Computer Graphics International CGI, IEEE Computer*], 520–527, Society Press (1998).

2. Conde, C. and Serrano, A., "3D Facial Normalization with Spin Images and Influence of Range Data Calculation over Face Verification," in [*Proc. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*], **16**, 115–120 (June 2005).

3. Bender, W., Gruhl, D., Morimoto, N., and Lu, A., "Techniques for Data Hiding," *IBM Systems Journal* **35**, 313–336 (February 1996).

4. Meerwald, P. and Uhl, A., "A Survey of Wavelet-Domain Watermarking Algorithms," in [*Proc. SPIE, Electronic Imaging, Security and Watermarking of Multimedia Contents III*], **4314**, 505–516, SPIE, IS&T, San Jose, CA, USA (January 2001).

5. Xia, X. G., Boncelet, C. G., and Arce, G. R., "A Multiresolution Watermark for Digital Images," in [*Proc. IEEE International Conference on Image Processing (IEEE ICIP 97)*], 548–551 (Oct. 1997).

6. Kundur, D. and Hatzinakos, D., "A Robust Digital Image Watermarking Scheme Using the Wavelet-Based Fusion," in [*Proc. IEEE International Conference on Image Processing (IEEE ICIP 97)*], **1**, 544–547 (Oct. 1997).

7. Piva, A., Bartolini, F., and Caldelli, R., "Self Recovery Authentication of Images in The DWT Domain," *Int. J. Image Graphics* **5**(1), 149–166 (2005).

8. Liu, J. L., Lou, D. C., Chang, M. C., and Tso, H. K., "A Robust Watermarking Scheme Using Self-Reference Image," *Computer Standards & Interfaces* **28**, 356–367 (2006).

9. Kong, X., Liu, Y., Liu, H., and Yang, D., "Object Watermarks for Digital Images and Video," *Image and Vision Computing* **22**, 583–595 (2004).

10. Noore, A., Singh, R., Vatsa, M., and Houck, M. M., "Enhancing Security of Fingerprints through Contextual Biometric Watermarking," *Forensic Science International* **169**(2–3), 188–194 (2007).

11. Uccheddu, F., Corsini, M., and Barni, M., "Wavelet-Based Blind Watermarking of 3D Models," in [*MM&Sec '04: Proceedings of the 2004 workshop on Multimedia and security*], 143–154, ACM, New York, NY, USA (2004).

12. Yu, Z., Ip, H. H. S., and Kwok, L. F., "A Robust Watermarking Scheme for 3D Triangular Mesh Models," *Pattern Recognition* **36**(11), 2603–2614 (2003).

13. Yin, K., Pan, Z., Shi, J., and Zhang, D., "Robust Mesh Watermarking Based on Multiresolution Processing," *Computers & Graphics* **25**(3), 409–420 (2001).

14. Hayat, K., Puech, W., and Gesquière, G., "Scalable 3D Visualization Through Reversible JPEG2000-Based Blind Data Hiding," *IEEE Trans. on Multimedia* **10**(7) (2008).

15. Shapiro, J., "Embedded Image Coding using Zerotrees of Wavelet Coefficients," *IEEE Trans. on Signal Processing* **41**(12), 3445–3462 (1993).