JPEG2000-Based Data Hiding and its Application to 3D Visualization

Khizar Hayat, William Puech, Gilles Gesquière

X

JPEG2000-Based Data Hiding and its Application to 3D Visualization

Khizar Hayat[a], William Puech[a] & Gilles Gesquière[b]
a.*LIRMM, UMR CNRS 5506, University of Montpellier II*
b.*LSIS, UMR CNRS 6168, Aix-Marseille University*
France

1. Introduction

This first decade of twenty-first century is witnessing revolution in the form of memory and network speeds as well as computing efficiencies. Simultaneously, the platform and client diversity base is also expanding, with powerful workstations on one extreme and handheld portable devices, like smart-phones on the other. The range of networks, even if we narrow it down, may end up at optical fiber to EDGE. Still versatile are the clients whose needs evolve with every passing moment. The technological revolution notwithstanding, dealing the accompanying diversity is always a serious challenge that must certainly require some scalable strategies. When it comes to scalability, one is compelled to think of the multi-resolution characteristic of wavelets as far as multimedia data is concerned. With the wavelets comes the JPEG2000 standard which has added newer dimensions to the concept of scalability. The scalability challenge is more glaring in the case of applications where disparate and large data is involved. An example application is the area of 3D visualization that requires more than one set of the data. Here, issues like data security, data authentication and data unification then emerge. All these form part and parcel of the field of data hiding. The emergence of the wavelet oriented JPEG2000 codec has brought with it the challenge of when, where and how to hide the data for watermarking or steganography while keeping in view the requirements of hiding capacity, robustness and imperceptibility. Even before the advent of the codec, wavelet transformations were commonplace in the field of signal processing. For the last few years, several techniques to embed data inside a host signal in the wavelet transform domain have been proposed. Few of the methods are compatible with the JPEG2000 standard and that is why one comes across methods exclusively proposed for the codec after the latter's inception. In today's world, data hiding is not limited to just watermarking, steganography or authentication but some non-traditional prospects are also there. One such area is the field of 3D visualization wherein data hiding can be employed as a tool for the synchronous and scalable unification of the disparate data in the JPEG2000 coding pipeline.

The rest of this chapter is arranged as follows. Section 2 is dedicated to the background concepts wherein a brief account of the wavelet transformations is being given, followed by the presentation of the salient features of the JPEG2000 codec. After introducing information

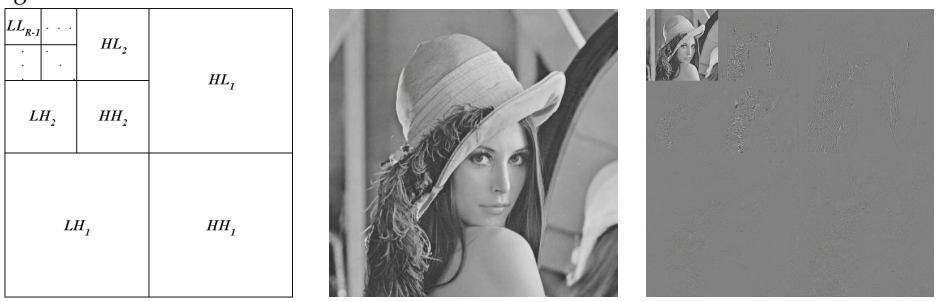
hiding concepts, Section 3 focuses on the when and where to embed in the JPEG2000 coding pipeline, which leads us to introduce a context level classification of the JPEG2000-based data hiding methods presented in the literature in Section 4. In Section 5 we present an application of the JPEG2000-based data hiding for a scalable and synchronous 3D visualization in a client/server environment. Section 6 sums up this chapter.

2. Discrete Wavelet Transform and JPEG2000

In standard image compression techniques, one of the essential steps is the domain transformation. The transform results in the decorrelation of the pixels. Pixel energy is thus compacted into a small number of coefficients. The idea is to enable the quantization step to trim the samples selectively, i.e. irrelevant samples must be quantized more heavily than the relevant ones [Taubman and Marcellin, 2002]. One aspect of the success of JPEG [Pennebaker and Mitchell, 1992] is in the efficiency of its transform, i.e. discrete cosine transformation (DCT) applied to each image block of 8 x 8. But the use of block-based transforms rests on the unrealistic assumption of the independence of blocks.

Of the alternatives available to the block-based transforms the one that got the approval of JPEG2000 [ISO/IEC, 2004] proponents is the discrete wavelet transform (DWT). The key being its multi-resolution nature resulting in sub-bands containing some level of detail derived from the whole image or at least a considerably large tile if the image is of very large size. The quality in such a situation is incremental as the lowest sub-band has the most important and relevant information and the higher sub-bands have finer details. Most of the energy is thus compacted into a few large transform coefficients -- an entropy coder easily locates these coefficients and encodes them.

DWT offers better energy compaction than the DCT without any blocking artifact after coding. In addition, the DWT decomposes the image into an L-level dyadic wavelet pyramid. The resultant wavelet coefficient can be easily scaled in resolution as one can discard the wavelet coefficients at levels finer to a given threshold and thus reconstruct an image with less detail.



(a) (b) (c)

Fig. 1. DWT; a) Sub-band structure, b) Lena, c) Lena at level 2 DWT decomposition.

The multi-resolution nature of DWT, therefore, makes it ideal for scalable image coding. After the color transformation, the DWT decomposes each component (luminance Y and chrominance Cr and Cb) into numerous frequency bands called sub-bands. For each level,

DWT is applied twice, once row-wise and once column-wise and hence four sub-bands result:

1. horizontally and vertically low-pass (LL),
2. horizontally low-pass and vertically high-pass (LH),
3. horizontally high-pass and vertically low-pass (HL) ,
4. horizontally and vertically high-pass (HH).

Let us consider the input image signal (or tile-component signal if image is large) as the LL_0 band. A (R−1)-level wavelet decomposition is associated with R resolution levels. Each sub-band of the decomposition is identified by its orientation (i.e. LL, LH, HL, and HH) and its corresponding decomposition level (0, 1, . . .R − 1). At each resolution level (except the lowest) the LL band is further decomposed. Thus the LL_0 band is decomposed to yield the LL_1, LH_1, HL_1 and HH_1 bands. Then, at the next level, as illustrated in Fig. 1.a, the LL_1 band is decomposed. This process is repeated until the LL_{R-1} band is obtained. If no transform is applied (R = 1) then there is only one sub-band: the LL0 sub-band. A level-2 wavelet decomposition of the image Lena, given in Fig. 1.b is illustrated in Figure. 1.c.

Right from the beginning, JPEG2000 has been supporting two kinds of transforms: the reversible integer-to-integer Daubechies (5/3) and the irreversible real-to-real Daubechies (9/7) [Daubechies and Sweldens, 1998, Sweldens, 1995] with the former being lossless and the latter being lossy. The 2D transformation can then be carried out by separately applying the 1D version horizontally and vertically one after the other. Let the 1D (pixel row or pixel column) input signal be $S_1,...S_n$ then for the reversible 5/3 wavelet transform the low-pass sub-band signal $L_1,...L_{n/2}$ and high-pass sub-band signal $H_1,...H_{n/2}$ are given by:

$$\begin{cases} H_i = S_{2i+1} - \left\lfloor \frac{1}{2}(S_{2i+2} + S_{2i}) \right\rfloor \\ L_i = S_{2i} + \left\lfloor \frac{1}{4}(H_i + H_{i-1}) + \frac{1}{2} \right\rfloor \end{cases} \quad (1)$$

For the lossy version we define a new set of variables $S'_1, \ldots S'_n$ where the odd numbered variables (S'_{2n+1}) will hold the first stage lifting outcomes and the even numbered (S'_{2n}) will hold those of second stage lifting. Let a, b, c and d are the first, second third and fourth stage parameters, respectively, then for bi-orthogonal 9/7 wavelet transform:

$$\begin{cases} S'_{2i+1} = S_{2i+2} + a.(S_{2i} + S_{2i+2}) \\ S'_{2i} = S_{2i} + b.(S'_{2i-1} + S'_{2i+1}) \\ H_i = \beta'.(S'_{2i+1} + c.(S'_{2i} + S'_{2i+2})) \\ L_i = \beta.(S'_{2i} + d.(H_{i-1} + H_i)), \end{cases} \quad (2)$$

where

$$a \approx -1.586134 \qquad b \approx -0.052980$$
$$c \approx 0.882911 \qquad d \approx 0.443506,$$

with β and β', being the scaling parameters, having the values:

$$\beta \approx 0.812893 \qquad \beta' \approx 1/\beta$$

The boundaries in both types of transforms are handled by utilizing symmetric extension[1].

In a simplified way, a typical JPEG2000 encoder[2] consists of the following steps [Taubman and Marcellin, 2002] which are illustrated in Fig. 2:

---

[1] http://research.microsoft.com/~jinl/paper_2002/msri_jpeg.htm

1. Preprocessing such as tiling and shifting the origin of the pixel values to 0 by subtracting 128.
2. Inter-component transform in the form of irreversible or reversible color transform to pass from RGB space to YCrCb space.
3. Intra-component transform that may be lossy or lossless DWT.
4. Quantization which decreases the size of the large coefficients and nullifies the small ones.
5. Tier 1 coding when the quantized coefficients are partitioned into rectangular code blocks and each is subjected independently to three coding passes. This step involves entropy coding too.
6. Tier 2 coding which is the packetization step whereby the code-pass data is converted to packets – these packets are combined to get the final image in the JPEG2000 format.

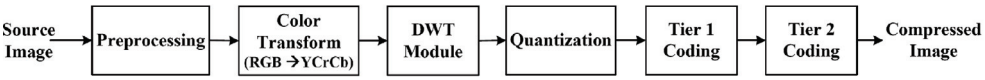


Fig. 2. A generalized scheme of the JPEG2000 encoder.

It must be noted that in a JPEG2000 coding pipeline there are two primary sources of data loss. One is obviously quantization and the other is the stage in tier-1 coding when a decision is made that which coding passes must be excluded from the final JPEG2000 file. For the application proposed in this chapter, the scalability prospects offered by JPEG2000 in the form of multi-resolution are to our advantage, especially in the client/server environment.

3. The When and Where of Information Hiding in JPEG2000

Data hiding deals with embedding information, called *message*, inside some host signal, like image, sound or video, called *cover* or *carrier*. The message may be small and robust 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 [Bender et al., 1996] characterize the effectiveness of a data hiding method, namely the *hiding capacity*, the *perceptual transparency*, the *robustness* and the *tamper resistance*. Hiding 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 very important. Domain-wise, embedding can be carried out in both the frequency domain and the transform domain. Pixel or coefficient allocation for data embedding may be regular (e.g. every k^{th} pixel or coefficient) 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

[2] http://www.ece.uvic.ca/~mdadams/jasper

embedding method is blind if data extraction by the recipient does not require the original cover.

Being an active research area for the last two decades, data hiding is now an established field and that is why a lot has been written about it [Cox et al., 2008]. We, therefore, focus on the literature about wavelet-based data hiding which is again very extensive and one is compelled to be brief and limit oneself to JPEG2000, as far as possible. Looking at the structure of JPEG2000 codec, as explained in Section 2, makes one think about when and where to interrupt the coding flow in order to embed the message. Theoretically, you can interrupt the codec anywhere for embedding but, at the periphery, the embedding capacity is lower, accompanied by relatively higher distortion.

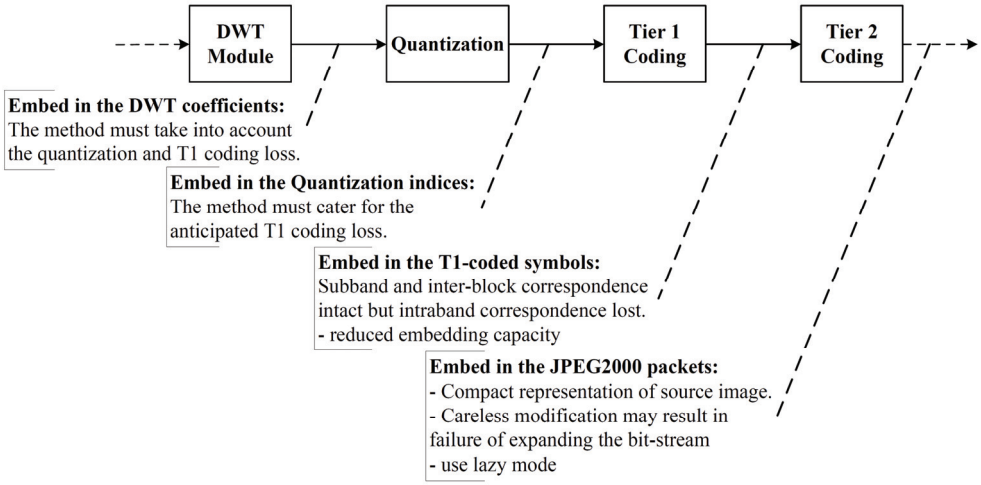


Fig. 3. Interrupting the JPEG2000 coding pipeline for information hiding.

Fig. 3 illustrates the potential interruption stages during the JPEG2000 coding to embed data in the to-be-encoded image[3]. Every type of intervention has its advantages and limitations.

- Embedding immediately after the DWT step would have the advantage of larger word size of the coefficients leading to high capacity. All the components are easily available one can allocate coefficients at will. This strategy may be especially convenient for JPEG2000 in lossless mode. The problem is however that steganalysis is easier since there is a high probability of unusual coefficient values. This is particularly true of coefficients belonging to high frequency sub-bands. Moreover embedding must be at least robust enough to resist the ensuing steps of quantization and T1-coding.
- Just after quantization, one can embed in the clipped coefficients with reduced capacity. The overhead of anticipating the loss, due to quantization, is eliminated with this type of embedding. Strictly speaking, however, the technique is the same as the last one and shares its pros and cons.
- As already stated T1-coding operates on the independence of blocks and comprises bit-plane coding with three passes in each bit-plane, namely significance,

[3] http://www.cs.nthu.edu.tw/~yishin

refinement and cleanup passes. This followed by the arithmetic coding (MQ coder). One way to intervene is to take advantage of the fact that the partitioned code blocks are coded independently using the bit-plane coder thus generating a sequence of symbols with some or all of these may be entropy coded. The T1 coded symbols from a given block vary in energy and the low index symbols are more energetic than the higher index ones. What can be done, for example, is to use the least energetic of these symbols, from the tail of the stream for each code block, for embedding implying non-random allocation. There is, however one problem in that the T1 coded symbols have smaller word size resulting in smaller embedding capacity and higher rate of distortion in quality as a result of embedding. This policy is not, however, advised in the lossless case since wordsizes of the coefficients are longer at the earlier steps thus leading to lesser distortions as result of embedding. In addition the embedding capacity is limited for such an embedding strategy and the rate of degradation is still larger.

An alternative approach could be to go for lazy mode and bypass arithmetic coding for most of the significance and refinement passes, except 4 MSBs, however. There would be no substantial benefit from entropy coding in such a scenario. The refinement pass carries subsequent bits after the MSB of each sample hence modification should not cause problems. The significant bits would act as masking which should make the modification of the subsequent bits less obvious. Hence the lazy mode mostly involves raw coding. Care must be taken in selecting potential raw coded magnitude refinement passes for embedding; otherwise there may be high degradation in quality. This may involve close examination of the bit-planes. The limitations are escalation in the size of the coded image and suspicion in the form of unusual bit stuffing and unusual appearance of error resilience marker.

- Subsequent to lazy mode encoding, one can also embed in the T2-coded bit-stream. This approach may be simple but has problems in the form of low capacity and high degradation wherein careless modification may result in failure of the expanding bit-stream. The easiest way for a novice may be to intervene here and that is why this intervention may be popular but this popularity makes it an easy target of steganalysis.

4. Context-Based Classification of JPEG2000 Data Hiding Methods

The wavelet-based information hiding can be classified in various ways depending on the criteria employed. Many criteria, like decomposition strategy, embedding technique, goal, application, extraction method and many others can be employed for classification. But for our purpose we will use classification where we will be taking into account the *when and where* factor to embed in the JPEG2000 coding pipeline. We call this a *context-based criterion* for classification. Before the advent of JPEG2000, many methods existed in the literature. A very elaborate compilation of these methods can be found in the form of [Meerwald, 2001a]. Not all of these methods are compatible with the JPEG2000 scheme. According to [Meerwald and Uhl, 2001], data hiding methods for JPEG2000 images must process the code blocks independently and that is why methods like inter-sub-band embedding [Kundur, 1999] and those based on hierarchical multi-resolution relationship [Kundur and

Hatzinakos, 1998] have not been recommended. In the same breath they reject the correlation-based method [Wang and Kuo., 1998] as well as non-blind methods. The reason for they give is the limited number of coefficients in a JPEG2000 code-block that are likely to fail in reliably detecting the hidden information in a single independent block.
The fact to classify in the context of JPEG2000 is driven by its coding structure as well as the multi-resolution character of DWT.

4.1 Embedding in the DWT coefficients

We further classify these methods into lowest sub-band methods, high or detail sub-band methods, trans-sub-band methods and methods exploiting the coefficient relationships in sub-band hierarchy.

4.1.1 Lowest sub-band methods

Embedding in lowest sub-band coefficient is suited for cases where the image has to be authenticated at every resolution level. The problem is however the size of the sub-band which is a dyadic fraction of the total, thus leading to reduced capacity. Moreover, since most of the energy is concentrated in the lowest sub-band, the embedding would definitely lead to low perceptual transparency. As an example of this type of embedding can be found in [Xiang and Kim, 2007] which uses the invariance of the histogram shape to rely on time-frequency localization property of DWT to propose a watermarking scheme that is resistant to geometric deformations. A geometrically invariant watermark is embedded into the low-frequency sub-band of DWT in such a way that the watermark is not only invariant to various geometric transforms, but also robust to common image processing operations.

4.1.2 High or detail sub-band methods

In contrast to low sub-bands, higher sub-bands may provide larger capacity. But this is accompanied by escalation in the final image size as the detail sub-band coefficients hover around zero. While explaining their method of embedding biometric data in fingerprint images, Noore *et al.* argue against the modification of the lowest sub-band to avoid degradation of the reconstructed image as most of the energy is concentrated in this band [Noore et al., 2007]. Instead they propose to redundantly embed information in all the higher frequency sub-bands. There are methods for embedding invisible watermarks by adding pseudo-random codes to large coefficients of the high and middle frequency bands of DWT but these methods have the disadvantage of being non-blind [Xia et al., 1997, Kundur and Hatzinakos, 1997]. An additive method transforms the host image into three levels of DWT and carry out embedding with the watermark being spatially localized at high-resolution levels [Suhail et al., 2003].

4.1.3 Inter sub-band methods

To avoid high computational cost for wavelet-based watermarking Woo *et al.* propose a simplified embedding technique that significantly reduces embedding time while preserving the performance of imperceptibility and robustness by exploiting implicit features of discrete wavelet transform (DWT) sub-bands, i.e. the luminosity information in the low pass band, and the edge information in the high pass bands [Woo et al., 2005]. The

method of Kong *et al.* embeds watermark in the weighted mean of the wavelets blocks, rather than in the individual coefficient, to make it robust and perceptually transparent [Kong et al., 2004]. One blind method transforms the original image by one-level wavelet transform and sets the three higher sub-bands to zero before inverse transforming it to get the modified image [Liu et al., 2006]. 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. The concept of Singular Value Decomposition (SVD) has been employed [Yavuz and Telatar, 2007] for their watermarking scheme wherein the $m \times n$ image matrix A is decomposed into a product of three matrices (USV^T); the $m \times m$ matrix U and $n \times n$ matrix V are orthogonal ($U^TU = I$, $V^TV = I$) and the $m \times n$ diagonal matrix S has r (rank of A) nonzero elements called singular values (SVs) of the matrix A. The SVs of the watermark are embedded into SVs of the LL and HL sub-bands of the cover image from level-3 DWT domain while components of U matrix of the watermark are embedded into LH and HH sub-bands. In extraction, first the similarity of extracted U matrix is checked with the original one. If it is found similar, the watermark is constructed by using extracted SVs and original U and V matrices of the watermark. Another DWT-SVD based method employs particle swarm optimizer (PSO) for watermarking [Aslantas et al., 2008]. Agreste *et al.* put forward a strong wavelet-based watermarking algorithm, called WM2.0 [Agreste et al., 2007]. WM2.0 embeds the watermark into high frequency DWT components of a specific sub-image and it is calculated in correlation with the image features and statistical properties. Watermark detection applies a re-synchronization between the original and watermarked image. The correlation between the watermarked DWT coefficients and the watermark signal is calculated according to the Neyman-Pearson statistic criterion just like the blind chaotic method of DWT oriented watermarking [Dawei et al., 2004]. The spread spectrum (SS) method by Maitya *et al.* embeds watermark information in the coefficients of LL and HH sub-bands of different decompositions [Maitya et al., 2007]. In two-band system, to increase embedding rate, the cover image is decomposed in different directions using biorthogonal wavelets (BiDWT). For embedding each watermark symbol bit, pseudo-random noise (PN) matrix of size identical to the size of LL sub-band coefficient matrix is generated and modulated by Hadamard matrix. This modulated code pattern is used to embed data in the LL sub-band while its bit-wise complement gives an orthogonal code pattern which is used for data embedding in the HH sub-band. To decode message bit for binary signaling, two correlation values (one from LL and the other from HH) are calculated. The overall mean of these correlation values serves as the threshold for watermark decoding.

4.1.4 Methods exploiting coefficient relationships in the sub-band coefficient hierarchy

Such methods may suitable for embedding resolution scalable messages. An example is image fusion when a small image is embedded in the larger one. Similarly 3D meshes can be embedded by hiding coarse meshes in low and finer details in high frequency coefficients. One can employ data structures like the embedded zero-tree wavelets (EZW [Shapiro, 1993]) or its improved form, the set partitioning in hierarchical trees (SPIHT [Said and Pearlman, 1996]). These structures enable to effectively remove the spatial redundancy across multi-resolution scales. The additional advantage is the provision of fine scalability. There is a method [Inoue et al., 1998] that exploits zero-tree structure by replacing the insignificant

coefficients with the addition/subtraction of small values. Uccheddu *et al.* adopt a wavelet framework in their blind watermarking scheme for 3D models under the assumption that the host meshes are semi-regular, thus paving the way for wavelet decomposition and embedding of the watermark at a suitable resolution level [Uccheddu et al., 2004]. 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.* 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 [Yu et al., 2003]. 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. A method adopts Guskov's multi-resolution signal processing method for meshes and uses a 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 [Yin et al., 2001]. The algorithm is integrable with the multi-resolution mesh processing toolbox and watermark detection requires registration and resampling to bring the attacked mesh model back into its original location, orientation, scale, topology and resolution level.
Besides above there may be methods involving specialized wavelets. Vatsa *et al.* present a 3-level redundant DWT (RDWT) biometric watermarking algorithm to embed the voice biometric Mel Frequency Cepstral (MFC) coefficients in a color face image of the same individual for increased robustness, security and accuracy [Vatsa et al., 2009]. Green channel is not used and after transforming the red and blue channels, watermarking is carried out followed by the inverse transform. Phase congruency model is used to compute the embedding locations which preserves the facial features from being watermarked and ensures that the face recognition accuracy is not compromised. The proposed watermarking algorithm uses adaptive user-specific watermarking parameters for improved performance. Yen and Tsai put forward an algorithm based on Haar DWT for the gray scale watermark by proposing visual cryptographic approach to generate two random shares of a watermark: one is embedded into the cover-image, another one is kept as a secret key for the watermark extraction later [Yen and Tsai, 2008].

4.2 Quantization-based methods

The authentication scheme described in [Piva et al., 2005] embeds an image digest in a subset of the sub-bands from the DWT domain. The image digest is derived from the DCT of the level 1 DWT *LL* sub-band of the image. The resultant DCT coefficients are scaled down by quantization and ordered from most to least significant through a zig-zag scan. A most significant subset, after discarding the DC coefficient, is quadruplicated for redundancy and then rescaled and scrambled by using two different keys. This gives the message which is substituted to the sub-bands selected from a set obtained by the further wavelet decomposition of the level 1 *HL* and *LH* sub-bands of the original image. Based on the significant difference of wavelet coefficient quantization, a blind algorithm groups every seven non-overlap wavelet coefficients of the host image into a block [Lin et al., 2008]. The two largest coefficients, in a given block, are referred to as significant coefficients and their difference as significant difference. The local maximum wavelet coefficient in a block is quantized by comparing the significant difference value in a block with the average

significant difference value in all blocks. The maximum wavelet coefficients are so quantized that their significant difference between watermark bit 0 and watermark bit 1 exhibits a large energy difference which can be used for watermark extraction. During the extraction, an adaptive threshold value is designed to extract the watermark from the watermarked image under different attacks. To determine the watermark bit, the adaptive threshold value is compared to the block-quantized significant difference. Jin *et al.* employ modulo arithmetic to constrain the noise resulted from the blind embedding into the quantized DWT coefficients directly. Ohyama *et al.* extract a least significant bit (LSB) plane of the quantized wavelet coefficients of the Y color component in a reversible way. They then embed the secret data and a JBIG2 bit-stream of a part of the LSB plane as well as the bit-depth of the quantized coefficients on some code-blocks [Ohyama et al., 2008]. Based on the compression ratio Li and Zhang propose an adaptive watermarking with the strength of watermark being proportional to the compression ratio to enable the embedded watermark survive the following code-stream rate allocation procedure without degrading the image quality [Li and Zhang, 2003].

There are methods that employ quantization index modulation (QIM). The idea is to quantize the host signal with a quantizer indexed by the message, i.e. if S is the embedded signal, M the message, and C the cover or host signal, then $S(C,M) = QM(C)$. The embedded signal should then be composed only of values in the set of quantizer outputs [Sullivan et al., 2004]. In the method of Ishida *et al.*, the QIM-JPEG2000 steganography, QIM is exploited with two different quantizers (one for embedding a '0' and other for a '1') to embed bit at the quantization step of DWT coefficients under the assumption that the probabilities of '0' and '1' are same in the message [Ishida et al., 2008]. A JPEG2000-based image authentication method employs extended scalar quantization and hashing for the protection of all the coefficients of the wavelet decomposition [Schlauweg et al., 2006]. The process involves feature extraction by wavelets to result in digital signature which, after encryption and error correction coding, is embedded as a removable watermark using the well-known QIM technique called dither modulation. The embedded watermark information is removable during the decompression process which is important for the improved image quality in the context of visualization. Traditionally, correlation analysis has been an integral part of the SS methods reported in various works - the principal difference being in the manner they ascertain the threshold for decoding.

4.3 Embedding in the compressed bit-stream

These methods usually involve partial or complete roll back of some coding steps, lazy mode coding. The blind scheme proposed in [Su et al., 2001] integrates data hiding with the embedded block coding with optimized truncation (EBCOT) and embed data during the formation of compressed bit stream. The method of Su and Kuo employs lazy coding to speed up the encoding process by skipping the 4 lowest bit planes during arithmetical encoding [Su and Kuo, 2003] . The authors maintain their software by the name stegoJasper, as reported in [Kharrazi et al., 2006] in which the bits are modified in function to their contribution in the reconstructed image at the decoder side, i.e. bits with least level of contributions are modified first. With this backward embedding approach they try to minimize the embedding artifact on the final embedded image. A similar method rolls back the JPEG2000 encoding process until the dequantization stage [Noda et al., 2003]. The method relies on the fact that the data has already passed the rate controller during the first

encoding and an aspired bitrate has already been established. Hence the second rate control should not be able to remove further information, so the additional information can be embedded after the quantization stage and the manipulated image data are again processed by the remaining parts of the JPEG2000 pipeline. To ensure the fidelity of the embedded data to further processing, the target bitrate may be set at a lower value for initial processing and set to the desired value for the second and final run. The technique is applicable during encoding as well as to already encoded JPEG2000 bit streams. One particular technique embeds watermark in the JPEG2000 pipeline after the stages of quantization and region of interest (ROI) scaling but before the entropy coding [Meerwald, 2001b]. A window sliding approach is adopted for embedding and for the sake of reliability the finest resolution sub-bands are avoided while the lowest frequencies carry higher payload.

5. An Application for Scalable Synchronized Surface-Based 3D Visualization

Volumes have been written on the traditional use of watermarking and steganography, in the form of copyrighting, authentication, security and many other applications. The JPEG2000 data hiding is not only valid for these as any generic technique but offers the additional advantage of multi-resolution to embed the message or the watermark in a scalable fashion. This aspect may have a particular value in the case, e.g. image fusion, where the message is not some plain text. We deviate, therefore, from the traditional course, to present a very interesting use of the JPEG2000 based data hiding in the field of surface-based 3D visualization.

5.1 Introduction

A typical 3D surface visualization is based on at least two sets of data: a 2D intensity image, called texture, with a corresponding 3D shape rendered in the form of a range image, a shaded 3D model and a mesh of points. A range image, also sometimes called a depth image, is an image in which the pixel value reflects the distance from the sensor to the imaged surface [Bowyer et al., 2006]. The underlying terminology may vary from field to field, e.g. in terrain visualization height/depth data is represented in the form of discrete altitudes which, upon triangulation, produce what is called a digital elevation model (DEM): the texture is a corresponding aerial photograph which is overlaid onto the DEM for visualization [Abdul-Rahman and Pilouk, 2008]. Similarly in 3D facial visualization the 2D color face image represents the texture but the corresponding depth map is usually in the form of what is called a 2.5D image. The latter is usually obtained by the projection of the 3D polygonal mesh model onto the image plane after its normalization [Conde and Serrano, 2005].

With the evolution of existing technologies, even if the quality of 3D visualization becomes very high, the client/server environments are very diverse in terms of network, computation and memory resources. Therefore, to cater each of the perspective clients, it is advisable to encode the data in a scalable way, unified into one standard format file. The JPEG2000 format offers the scalability thanks to the multi-resolution nature of its discrete wavelet transform (DWT). For the integration of all the data into one file one can rely on the technique of data hiding due to the smaller size of the depth map file as it can be embedded in the bulky texture image. But this embedding must be carried out in such a way that the JPEG2000 file format is conserved. In addition, the embedding must not interfere with the

multi-resolution hierarchy of the JPEG2000. As a consequence, for each of the possible resolutions, the corresponding texture and its depth map must be recoverable at the decoder.
In this section, the synchronized unification of the range data with the corresponding texture is realized by the application of perceptually transparent DWT domain data hiding strategies. In order to conserve the high quality of visualization we are relying on the LSB-based embedding. At the beginning we interrupt immediately after the DWT stage for embedding but then discuss the prospects of some other type of interventions too. The proposed methods are blind in the sense that only a secret key, if any and the size of the range image are needed to extract the data from the texture image.

5.2 The proposed strategy

A precursor of this method can be found in [Hayat et al., 2008b] 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 due to considerable disparity between the texture and its depth map in the context of size. For the work in perspective we have chosen the worst case scenario, i.e. same size of texture and the depth map. This should have an additional advantage to have a clearer idea of the embedding capacity. As a case study we are taking a 3D face visualization example.

5.2.1 Background

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, recreating a simulation replica of a real meeting room by visualizing the "virtual heads" around a table [Weik et al., 1998]. 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[4], 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 user's 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 technologies 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 video-

[4] www.tixeo.com

surveillance 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.

The objective over here is to reduce the data considerably for optimal real-time 3D facial visualization in a client/server environment. As already stated, 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. For 3D visualization one would thus have to manipulate at least two files. It would be better to have a single file rather than two. We propose to unify the two files into a single standard JPEG2000 format file. The use of DWT-based JPEG2000 will give us two specific advantages, aside from the compression it offers. One, the multi-resolution 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 transmitting 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.

5.2.2 The embedding step

For an $N \times N$ pixel facial texture and its corresponding $M \times M$ point depth map (2.5D) we propose our data hiding strategy presented in Fig. 4. 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 carries $\lfloor k \rfloor$ 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 especially true when $M < N/3$ as one has the facility to run a pseudo-random generator (PRNG) to select the potential carrier coefficients.

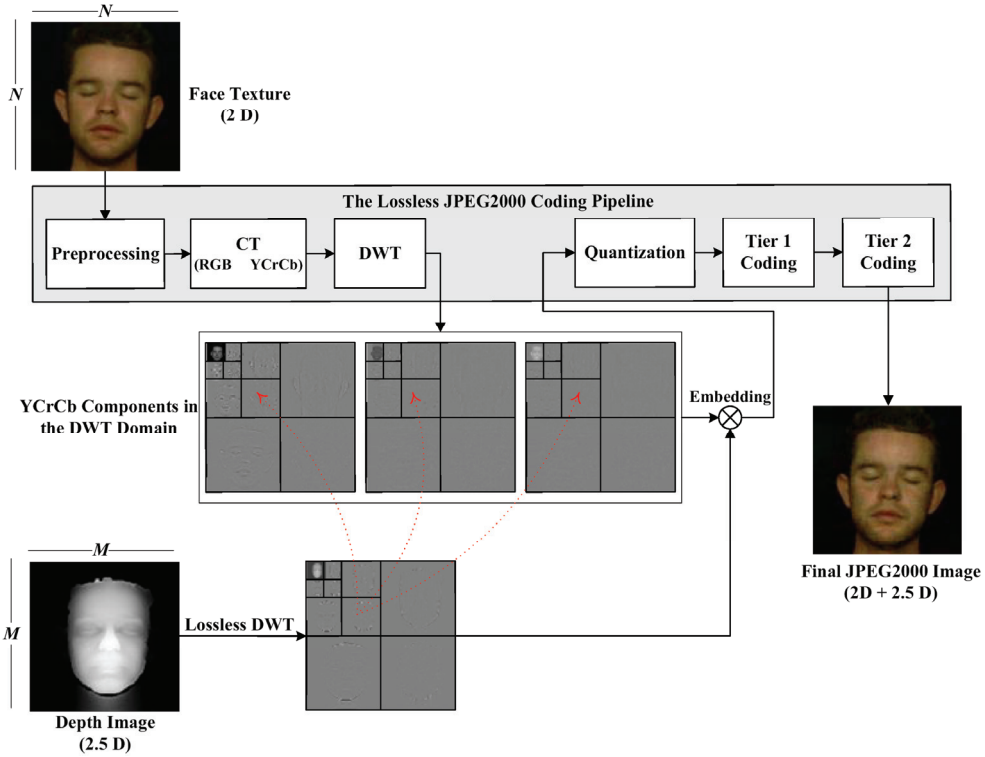


Fig. 4. Description of the method.

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 components are re-inserted into the JPEG2000 coding pipeline. The result is a monolithic JPEG2000 format face texture image that has the depth map hidden in it. A raw description of the embedding strategy is outlined in Algorithm 1. The use of nested loop may be misleading for some readers but it must be borne in mind that the loops are finite and does not imply by any means a cubic complexity. We have written this algorithm for the sake of comprehension.

Algorithm 1 The embedding strategy.

begin
read the level-L DWT-domain carrier texture C from the JPEG2000 coding pipeline, such that $C \subseteq \{Y, Cr, Cb\}$
read the corresponding depth map file
apply lossless DWT to the depth map at level L to get M and optimize if needed
initialize a PRNG with a seed K
for each subband of C and its counterpart in M **do**
  partition the C subband into square blocks in relation to the M subband
  for each correspondence of C subband block $B_{u,v}$ and its M subband k-bit coefficient $d_{u,v}$ **do**
    for $p = 0$ to k **do**
      generate the index which is the next pseudo-random number to get i, j
      get the C subband coefficient $t_{i,j}$
      substitute the LSB of $t_{i,j}$ with the p^{th} bit of $d_{u,v}$
    end for
  end for
end for
label the block-wise marked C as C'
end

5.2.3 Optimization in embedding

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 sub-bands. Hence one can always use a flag bit to differentiate this case from the rest. In addition, the use of k^{th} 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* ϵ [−128, 127] then concatenate *coeff* to 0 and embed as 9bits;
- **else if** *coeff* = 0 then embed binary 11;
- **else if** *coeff* ϵ [−192,−128] then concatenate |−128 − *coeff*| to 101 & 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.

5.2.4 Decoding and reconstruction

Just before the inverse DWT stage of the JPEG2000 decoder, the DWT domain depth map can be blindly extracted by reversing the embedding process mentioned above. In the reconstruction phase, by the application of 0-*padding*, one can have L+1 different approximation images of facial texture/depth map pair. And this is where one can achieve the scalability goal. Our method is based on the fact that it is not necessary that all the data is available for reconstruction. This is one of the main advantages of the method since the depth map and facial texture can both be reconstructed with even a small subset of the transmitted carrier coefficients. The resolution scalability of wavelets and the synchronized character of our method enable a 3D visualization even with fewer than original resolution layers as a result of partial or delayed data transfer. The method thus enables to effect visualization from a fraction of data in the form of the lowest sub-band, 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 sub-bands at level L', with $L' \leq L$. For the rest of $3L'$ parts one can always pad a 0 for each of their coefficient as shown in Algorithm 2. The inverse DWT of the 0-stuffed transform components will yield what is known as image of approximation of level L0. 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 $3(L-L')+1$ sub-bands. 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 on 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 $3L'$sub-bands of transformed DEM before inverse DWT that would result in a level L'-approximate DEM of an inferior quality.

Algorithm 2 The reconstruction process via 0-padding.

1: **begin**
2: **read** the coded texture data corresponding to level-L'' subbands
3: **decode** to extract the range data which may correspond to L'' or a larger level $L'' + \Delta$, where $\Delta \geq 0$ depends on the extent of adaptation
4: **for** both the texture and the extracted range data **do**
5: **initialize** c_l to L'' if dealing with texture or to $L'' + \Delta$, otherwise
6: **stuff** a '0' for every coefficient of the subbands with level$\neq c_l$
7: **apply** inverse DWT to the result
8: **add** the result with the $(c_l - 1)$ approximation to get the (c_l) approximation
9: **end for**
10: **end**

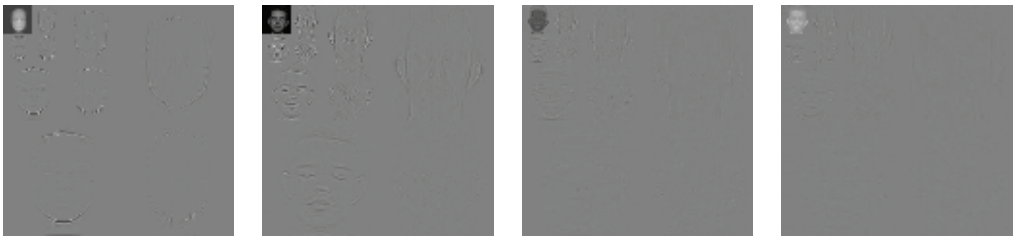
5.3 Example Simulation

We have applied our method to a number of examples from FRAV3D[5] database. One such example is given in that consists of a 120×120 point 2.5D depth map (Fig. 5.a) corresponding to a 120 × 120 pixel colored 2D face image given n Fig. 5.b. Each point of the 2.5D depth map is coded with 8 bits. A 3D visualization based on the two images is depicted by a view given in Fig. 5.c. Lossless DWT is applied in isolation to the depth map at level-3 to get the image given in Fig. 6.a. To ensure the accuracy we represent each of the transformed depth map coefficients 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. 6.b-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. 5. Original data: a) 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)

[5] http://www.frav.es/databases/FRAV3d



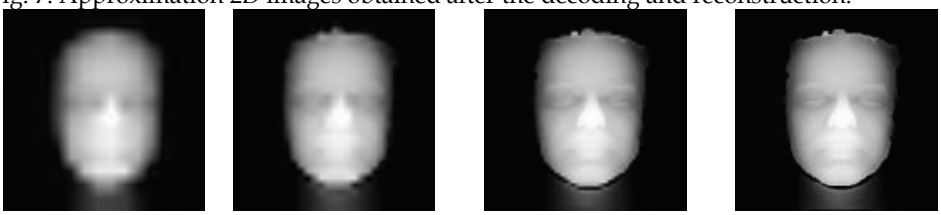
(a) 2.5D (b) Y (c) Cr (d) Cb

Fig. 6. Level-3 DWT domain images: a) depth map, b-d) components of the transformed 2D face image from the lossless JPEG2000 coding pipeline.

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 to the available lowest frequency $3(L\text{-}L') + 1$ sub-bands. 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 sub-bands involved being 1, 4, 7 and 10 out of the total of 10 sub-bands, respectively. For visual comparison, the approximation 2D images are given in Fig. 7 while the approximation depth maps are shown in Fig. 8.

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

Fig. 7. Approximation 2D images obtained after the decoding and reconstruction.



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

Fig. 8. Approximation 2.5D images obtained after the decoding and reconstruction.

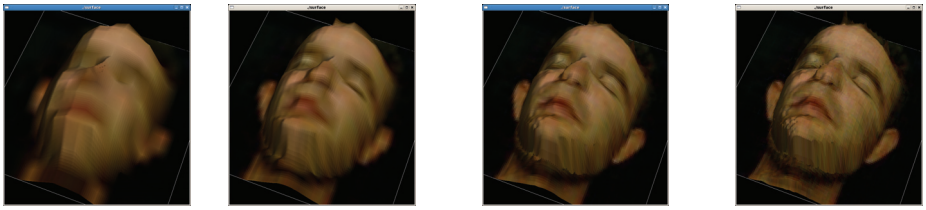
For the purpose of quantitative comparison the mean results over all the FRAV3D 2*D*/2.5*D* pairs subjected to our method are tabulated in the form of Table 1. **Results obtained for 2D face image after the extraction and reconstruction as a function of the transmitted data.** and Table 2.

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. Results obtained for 2D face image after the extraction and reconstruction as a function of the transmitted data.

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
MSE	11.33	7.76	4.73	0
PSNR (*dB*)	27.05	30.33	34.63	∞

Table 2 Results obtained for the depth map after the extraction and reconstruction as a function of the transmitted data.



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

Fig. 9. 3D views from the visualization with different 2D/2.5D Approximation pairs

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% 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 PSNRs in the order of 45 *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 5.4. 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 sub-band, the bitrate stays at 0.14 for both. We have used root mean square error (RMSE) as an error measure in length units for 2.5*D*. The 3D visualization obtained from the approximation 2*D*/2.5*D* pairs is depicted in the form of a 3D view at a particular angle in Fig. 9.

5.4 Innovations

We have applied this work to many examples of terrain visualization without the optimization step [Hayat et al., 2008b]. Taking another terrain case study we changed the codec to the lossless mode and interrupted after the quantization stage [Hayat et al., 2008a]. In T1 coding, which is the first of the two coding stages of JPEG2000, the quantizer indices for each sub-band are partitioned into rectangular code blocks with its nominal dimensions being dyadic and their product not exceeding 4096. The partitioned code blocks are coded independently using the bit-plane coder thus generating a sequence of symbols with some or all of these may be entropy coded. Due to this independent encoding of code blocks, the correspondence between the lossless DWTed DEM and lossy DWTed Y plane of texture is maintainable. The T1 coded symbols from a given block vary in energy and the low index

symbols are more energetic than the higher index ones. What we do is to use the least energetic of these symbols, from the tail of the stream for each code block, for LSB embedding implying non-random allocation. There is, however one problem in that the T1 coded symbols have smaller word-size resulting in smaller embedding capacity and higher rate of distortion in quality as a result of embedding. This policy is not, however, advised in the lossless case since word-sizes of the coefficients are longer at the earlier steps thus leading to lesser distortions as result of embedding. Hence one can embed immediately after the DWT step at the earliest.

Another possible innovation is to adapt the synchronization in the method by employing lower level of DWT for the decomposition of the depth map due to the latter's critical nature. That would imply that, rather than utilizing all the texture sub-bands for embedding, utilize a subset on the low frequency side. For the details of this strategy, with an LSB based embedding and for a terrain example, consult [Hayat et al., 2008c].

In all of the above, LSB based embedding technique has been employed which makes these technique perceptually transparent but at the cost of robustness. A spread spectrum technique may improve the robustness. An adaptive strategy with SS embedding has been presented in [Hayat et al., 2009]. The additional advantage of this strategy is its highest possible imperceptibility due to its removable nature.

6. Summary

We started this chapter with a brief description of the DWT with a focus on the Daubechies' algorithms supported by the JPEG2000 codec. This was followed by a stepwise introduction to the structure of the JPEG2000 encoding. The structure served as the foundation to analyse various stages where one can embed the data to be hidden. This analysis led us to classify, on the basis of context, the large number of JPEG2000-based embedding methods presented in the literature in the past few years. In the end, a novel application of the JPEG2000-based information hiding, for synchronized and scalable 3D visualization, was presented.

In a nutshell, this chapter not only presented a non-traditional application of wavelet based data hiding but also a detailed, though compact, survey of the state-of-the-art techniques in this context. Our approach to classify the methods on the base of context must be helpful to readers to understand recent JPEG2000 data hiding approaches. Coming back to the non-traditional application, the results of our simulation have been interesting in the sense that even with a depth map of the same dimensions as the 2D face image one got 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 from 35.09 dB to a maximum of 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 videoconferencing and video-surveillance applications 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.

7. References

[Abdul-Rahman and Pilouk, 2008] A. Abdul-Rahman and M. Pilouk. *Spatial Data Modelling for 3D GIS*. Springer, 2008.

[Agreste *et al.*, 2007] S. Agreste, G. Andaloro, D. Prestipino, and L. Puccio. An Image Adaptive, Wavelet-Based Watermarking of Digital Images. *Journal of Computational and Applied Mathematics*, 210(1–2):13–21, 2007.

[Aslantas *et al.*, 2008] V. Aslantas, A .L. Dogan, and S. Ozturk. DWT-SVD Based Image Watermarking Using Particle Swarm Optimizer. In *Proc. ICME'08, IEEE International Conference on Multimedia & Expo*, pages 241–244, June 2008.

[Bender *et al.*, 1996] W. Bender, D. Gruhl, N. Morimoto, and A. Lu. Techniques for Data Hiding. *IBM Systems Journal*, 35(3-4):313–336, February 1996.

[Bowyer *et al.*, 2006] K. W. Bowyer, K. Chang, and P. Flynn. A Survey of Approaches and Challenges in 3D and Multi-modal 3D + 2D Face Recognition. *Computer Vision & Image Understanding*, 101(1):1–15, 2006.

[Conde and Serrano, 2005] C. Conde and A. Serrano. 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*, volume 16, pages 115–120, June 2005.

[Cox *et al.*, 2008] I. J. Cox, M. L. Miller, and J. A. Bloom. *Digital Watermarking*. Morgan Kaufmann Publishers, 2008.

[Daubechies and Sweldens, 1998] I. Daubechies and W. Sweldens. Factoring Wavelet Transforms into Lifting Steps. *Fourier Anal. Appl.*, 4(3), 1998.

[Dawei *et al.*, 2004] Z. Dawei, C. Guanrong, and L. Wenbo. A Chaos-Based Robust Wavelet-Domain Watermarking Algorithm. *Chaos, Solitons & Fractals*, 22(1):47–54, 2004.

[Hayat *et al.*, 2008a] K. Hayat, W. Puech, and G. Gesquière. A Lossy JPEG2000-Based Data Hiding Method for Scalable 3D Terrain Visualization. In *Proc. EUSIPCO'08: the 16th European Signal Processing Conference*, Lausanne, Switzerland, August 2008.

[Hayat *et al.*, 2008b] K. Hayat, W. Puech, and G. Gesquière. Scalable 3D Visualization Through Reversible JPEG2000-Based Blind Data Hiding. *IEEE Trans. Multimedia*, 10(7):1261–1276, November 2008.

[Hayat *et al.*, 2008c] K. Hayat, W. Puech, and G. Gesquière. Scalable Data Hiding for Online Textured 3D Terrain Visualization. In *Proc. ICME'08, IEEE International Conference on Multimedia & Expo*, pages 217–220, June 2008.

[Hayat *et al.*, 2009] K. Hayat, W. Puech, and G. Gesquière. An Adaptive Spread Spectrum (SS) Synchronous Data Hiding Strategy for Scalable 3D Terrain Visualization. In *Proc. SPIE, Electronic Imaging, Visualization and Data Analysis*, volume 7243, San Jose, CA, USA, January 2009.

[Inoue *et al.*, 1998] H. Inoue, A. Miyazaki, A. Yamamoto, and T. Katsura. A digital watermark based on the wavelet transform and its robustness on image compression. In *Proc. ICIP 98. the 1998 International Conference on Image Processing*, volume 2, pages 391–395 vol.2, October 1998.

[Ishida *et al.*, 2008] T. Ishida, K. Yamawaki, H. Noda, and M. Niimi. Performance Improvement of JPEG2000 Steganography Using QIM. In *Proc. IIH-MSP '08: the 2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pages 155–158, Washington, DC, USA, 2008.

[ISO/IEC, 2004] ISO/IEC. *ISO/IEC 15444-1: Information Technology, JPEG2000 Image Coding System, Part 1: Core Coding System*. ISO Central Secretariat: CH-1211 Geneva, Switzerland, 2004.

[Kharrazi *et al.*, 2006] M. Kharrazi, H. T. Sencar, and N. Memon. Performance Study of Common image Steganography and Steganalysis Techniques. *Journal of Electronic Imaging*, 15(4):041104, 2006.

[Kong *et al.*, 2004] X. Kong, Y. Liu, H. Liu, and D. Yang. Object Watermarks for Digital Images and Video. *Image and Vision Computing*, 22:583–595, 2004.

[Kundur and Hatzinakos, 1997] D. Kundur and D. Hatzinakos. A Robust Digital Image Watermarking Scheme Using the Wavelet-Based Fusion. In *Proc. IEEE International Conference on Image Processing (IEEE ICIP 97)*, volume 1, pages 544–547, Santa Barbara, CA, USA, October 1997.

[Kundur and Hatzinakos, 1998] D. Kundur and D. Hatzinakos. Digital Watermarking Using Multiresolution Wavelet Decomposition. In *Proc. IEEE International Conference on Acoustic, Speech and Signal Processing (IEEE ICASSP 98)*, volume 5, pages 2969–2972, Seattle, WA, USA., May 1998.

[Kundur, 1999] D. Kundur. Improved Digital Watermarking Through Diversity and Attack Characterization. In *Proc. ACM Workshop on Multimedia Security'99*, pages 53–58, Orlando, FL, USA, October 1999.

[Li and Zhang, 2003] K. Li and X. P. Zhang. Reliable Adaptive Watermarking Scheme Integrated with JPEG2000. In *Proc. of 3rd International Symposium on Image and Signal Processing and Analysis (ISPA 2003)*, Rome, Italy, September 2003.

[Lin *et al.*, 2008] W.-H. Lin, S.-J. Horng, T.-W. Kao, P. Fan, C.-L. Lee, and Y. Pan. An Efficient Watermarking Method Based on Significant Difference of Wavelet Coefficient Quantization. *IEEE Trans. Multimedia*, 10(5):746–757, 2008.

[Liu *et al.*, 2006] J. L. Liu, D. C. Lou, M. C. Chang, and H. K. Tso. A Robust Watermarking Scheme Using Self-Reference Image. *Computer Standards & Interfaces*, 28:356–367, 2006.

[Maitya *et al.*, 2007] S. P. Maitya, M. K. Kundub, and T. S. Das. Robust SS Watermarking with Improved Capacity. *Pattern Recognition Letters*, 28(3):350–356, 2007.

[Meerwald and Uhl, 2001] P. Meerwald and A. Uhl. A Survey of Wavelet-Domain Watermarking Algorithms. In *Proc. SPIE, Electronic Imaging, Security and Watermarking of Multimedia Contents III*, volume 4314, pages 505–516, San Jose, CA, USA, January 2001.

[Meerwald, 2001a] P. Meerwald. Digital Image Watermarking in the Wavelet Transform Domain. Master's thesis, Department of Scientific Computing, University of Salzburg, Austria, January 2001.

[Meerwald, 2001b] P. Meerwald. Quantization Watermarking in the JPEG2000 Coding Pipeline. In *Proc. Communications and Multimedia Security Issues of The New Century, IFIP TC6/TC11 Fifth Joint Working Conference on Communications and Multimedia Security, CMS '01*, pages 69–79, May 2001.

[Noda *et al.*, 2003] H. Noda, J. Spaulding, M. N. Shirazi, M. Niimi, and E. Kawaguchi. Bit-plane decomposition steganography combined with jpeg2000 compression. In *IH '02: Revised Papers from the 5th International Workshop on Information Hiding*, pages 295–309, London, UK, 2003.

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

[Ohyama *et al.*, 2008] S. Ohyama, M. Niimi, K. Yamawaki, and H. Noda. Reversible data hiding of full color jpeg2000 compressed bit-stream preserving bit-depth information. In *Proc. ICPR'08: the 19th International Conference on Pattern Recognition*, pages 1–4, December 2008.

[Pennebaker and Mitchell, 1992] W. B. Pennebaker and J. L. Mitchell. *JPEG: Still Image Data Compression Standard*. Springer, 1992.

[Piva *et al.*, 2005] A. Piva, F. Bartolini, and R. Caldelli. Self Recovery Authentication of Images in the DWT Domain. *Int. J. Image Graphics*, 5(1):149–166, 2005.

[Said and Pearlman, 1996] A. Said and W.A. Pearlman. A new, fast, and efficient image codec based on set partitioning in hierarchical trees. *Circuits and Systems for Video Technology, IEEE Transactions on*, 6(3):243–250, June 1996.

[Schlauweg *et al.*, 2006] M. Schlauweg, D. Pröfrock, and E. Müller. JPEG2000-Based Secure Image Authentication. In *Proc. MM&Sec '06: the 8th Workshop on Multimedia and Security*, pages 62–67, New York, NY, USA, 2006.

[Shapiro, 1993] J.M. Shapiro. Embedded Image Coding using Zerotrees of Wavelet Coefficients. *IEEE Trans. Signal Processing*, 41(12):3445–3462, 1993.

[Su and Kuo, 2003] P. C. Su and C .-C .J. Kuo. Steganography in jpeg2000 compressed images. *IEEE Trans. Consumer Electronics*, 49(4):824–832, November 2003.

[Su *et al.*, 2001] P. C. Su, H. J. Wang, and C. C. J. Kuo. An Integrated Approach to Image Watermarking and JPEG-2000 Compression. *Journal of VLSI Signal Processing Systems, Special Issue on Multimedia Signal Processing*, 27(1-2):35–53, June 2001.

[Suhail *et al.*, 2003] M. A. Suhail, M. S. Obaidat, S. S. Ipson, and B. Sadoun. A Comparative Study of Digital Watermarking in JPEG and JPEG 2000 Environments. *Inf. Sci. Inf. Comput. Sci.*, 151:93–105, 2003.

[Sullivan *et al.*, 2004] K. Sullivan, Z. Bi, U. Madhow, S. Chandrasekaran, and B.S. Manjunath. Steganalysis of quantization index modulation data hiding. In *IEEE International Conference on Image Processing*, pages 1165–1168, October 2004.

[Sweldens, 1995] W. Sweldens. The Lifting Scheme: a New Philosophy in Biorthogonal Wavelet Constructions. In *Proc. SPIE, Electronic Imaging, Wavelet Applications in Signal and Image Processing*, volume 2569, pages 68–79, San Jose, CA, USA, September 1995.

[Taubman and Marcellin, 2002] D. S. Taubman and M. W. Marcellin. *JPEG2000: Image Compression Fundamentals; Standards and Practice*. Springer, 2002.

[Uccheddu *et al.*, 2004] F. Uccheddu, M. Corsini, and M. Barni. Wavelet-Based Blind Watermarking of 3D Models. In *Proc. MM&Sec '04: Workshop on Multimedia and Security*, pages 143–154, New York, NY, USA, 2004.

[Vatsa *et al.*, 2009]M. Vatsa, R. Singh, and A. Noore. Feature Based RDWT Watermarking for Multimodal Biometric System. *Image and Vision Computing*, 27(3):293–304, 2009.

[Wang and Kuo., 1998] H. J. Wang and C. C. Kuo. An Integrated Approach to Embedded Image Coding and Watermarking. In *Proc. IEEE International Conference on Acoustic, Speech and Signal Processing (IEEE ICASSP 98)*, pages 3271–3275, Seattle, WA, USA, May 1998.

[Weik *et al.*, 1998] S. Weik, J. Wingbermuhle, and W. Niem. Automatic Creation of Flexible Antropomorphic Models for 3D Videoconferencing. In *Proc. CGI'98: Computer Graphics International*, pages 520–527, 1998.

[Woo *et al.*, 2005] C. S. Woo, J. Du, and B. Pham. Performance Factors Analysis of a Wavelet-Based Watermarking Method. In *Proc. ACSW Frontiers '05: the Australasian Workshop on Grid Computing and e-Research*, pages 89–97, Darlinghurst, Australia, 2005.

[Xia *et al.*, 1997] X. G. Xia, C. G. Boncelet, and G. R. Arce. A Multiresolution Watermark for Digital Images. In *Proc. IEEE International Conference on Image Processing (IEEE ICIP 97)*, pages 548–551, Santa Barbara, CA, USA, October 1997.

[Xiang and Kim, 2007] S. Xiang and H. J. Kim. Geometrically Invariant Image Watermarking in the DWT Domain. In *WISA*, volume 4867 of *Lecture Notes in Computer Science*, pages 76–90, 2007.

[Yavuz and Telatar, 2007] E. Yavuz and Z. Telatar. Improved SVD-DWT Based Digital Image Watermarking Against Watermark Ambiguity. In *Proc. SAC '07: the ACM Symposium on Applied Computing*, pages 1051–1055, New York, NY, USA, 2007.

[Yen and Tsai, 2008] E. Yen and K. S. Tsai. HDWT-Based Grayscale Watermark for Copyright Protection. *Expert Systems with Applications*, 35(1–2):301–306, 2008.

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

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