Conference or Workshop Item:
Bhowmik, D. and Abhayaratne, C. (2012) Distortion constrained robustness scalable watermarking. In: Image Processing (IPR 2012), IET Conference on , 03-04 Jul 2012, London, UK.

https://doi.org/10.1049/cp.2012.0434

Distortion constrained robustness scalable watermarking

Deepayan Bhowmik and Charith Abhayaratne

Department of Electronic and Electrical Engineering, The University of Sheffield,
Sheffield, United Kingdom, S1 3JD
{d.bhowmik, c.abhayaratne}@sheffield.ac.uk

Keywords: Scalable watermarking, embedding distortion, robustness, JPEG 2000.

Abstract

The embedding distortion and the robustness to quality scalable image coding are two complementary watermarking requirements. This work proposes a novel concept of scalable image watermarking to generate a distortion-constrained robustness scalable watermarked image code stream which consists of hierarchically nested joint distortion-robustness coding atoms. The code stream is generated using a new wavelet domain binary tree guided rules-based blind watermarking algorithm. The code stream can be truncated at any distortion-robustness atom level to generate the watermarked image with the desired distortion-robustness requirements. A universal blind extractor is capable of extracting watermark data from the watermarked images. The simulation results verify the feasibility of the proposed concept, its applications and its improved robustness to quality scalable content adaptation (JPEG 2000).

1 Introduction

Although a wide variety of watermarking schemes have been offered to date, a fundamentally traditional concept is still followed in almost all the schemes. In this paper, we propose a novel concept of scalable watermarking as opposed to traditional watermarking schemes, by creating hierarchically nested joint distortion-robustness coding atoms. The watermarking scalability here refers to embed the watermarks in a hierarchical fashion in such a way that more embedding information leads to better robustness. The concept of scalable watermarking is particularly useful in watermarking for scalable coded images where the watermark can also be scaled according to the heterogenous network capacity and the end user's requirement for a target application. For example, for a high bandwidth network and a high resolution display, highly imperceptible but less robust watermarked image can be transmitted. As in this scenario, high quality media is desirable and the watermark can also be extracted reliably due to lesser compression. Whereas, for a low capacity network and low resolution display, the distribution server can choose highly robust watermarking stream, where, due to higher compression the watermarking imperceptibility is less important but high robustness is required for a reliable watermark extraction (refer Fig. 7). Similarly, based on any other combinations of the network's capability and user's requirement, the scalable watermarked media code stream can be truncated and distributed accordingly.

With the increased use of scalable coded media, such a scalable watermarking concept is very important, but a little or no work has been proposed so far in the current literature. Available most common such algorithms are proposed either as a joint progressive scalable watermarking and coding scheme [2, 5] or efficient coefficient selection methods which are robust against resolution or quality scalable attacks [1, 3]. These algorithms primarily focused on two main robustness issues [4]: 1) detection of watermark after acceptable scalable compression and 2) graceful improvement of extracted watermark as more quality or resolution layers are received at the image decoder.

On the contrary, in this paper we propose a novel scalable watermarking concept, using a distortion constrained watermarked code stream to generate a watermarked image with desired distortion robustness requirements. The contributions of the proposed scheme, are:

– Scalable embedded code stream generation using hierarchically nested joint distortion-robustness coding atoms.

– The code stream should be allowed to be truncated at any distortion-robustness atom level to generate the watermarked image with the desired distortion-robustness requirements.

To achieve these goals, we introduced a new wavelet domain binary tree guided rules-based blind watermarking algorithm. The universal blind extractor is capable of extracting watermark data from the watermarked images, created using any truncated code stream.

2 The Proposed Scalable watermarking

2.1 The embedding algorithm

To satisfy the requirements of scalable watermarking, we introduce a new watermarking algorithm which creates watermarked image code stream atoms and allows quantitative embedding distortion measurement at individual atom level. The watermark embedding is performed on the wavelet coefficients generated after the forward transformation. The embedding algorithm follows a non-uniform quantization based index modulation as described below. The embedding process is divided into three parts: 1) Quantized binary tree formation, 2) Embedding and 3) Extraction & authentication.

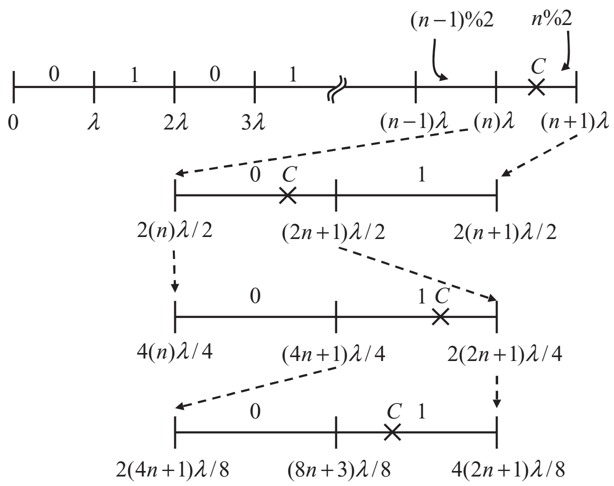

Figure 1: The hierarchical quantizer in formation of the binary tree.

2.1.1 Quantized binary tree formation

The individual coefficients are recursively quantized to form a binary tree. The coefficients to be watermarked, can be selected based on their magnitude, sign, texture information, randomly or any other selection criteria. Firstly the selected coefficient (C) is indexed (b_i) as 0 or 1 using an initial quantization parameter λ:

$$b_i = \left\lfloor \frac{|C|}{\lambda/2^i} \right\rfloor \%2, \;\; i \in 0,1,2,3..., \tag{1}$$

where $\%$ denotes the modulo operation. Assuming $n = \left\lfloor \frac{|C|}{\lambda} \right\rfloor$, we can identify the position of C between the quantized cluster (n) and $(n+1)$ as shown in Fig. 1. The coefficient C is then further quantized more precisely within a smaller cluster using a smaller quantization parameter $\lambda/2$ and corresponding index is calculated as: $b_1 = \left\lfloor \frac{|C|}{\lambda/2} \right\rfloor \%2$. The index tree formation is continued recursively by scaling λ by 2, until $\lambda/2^i \geq 1$. During this tree formation process the *Sign* of the coefficients is preserved separately.

Now based on the calculated index value at various quantization steps, a binary tree ($b(C)$) of each coefficient is formed as follows:

$$b(C) = (b_0)(b_1)....(b_{i-1})(b_i), \tag{2}$$

where $(b_0), (b_1)...(b_i)$ are binary bits with b_0 as most significant bit (MSB) and b_i as least significant bit (LSB) of the binary tree along with the tree depth of $d = i+1$. For example if $C = 135$ and initial $\lambda = 30$, the binary tree $b(C)$ will be $b(C)$ = 01000 and d=6.

2.1.2 Embedding

The binary tree is used to embed the binary watermark information using symbol based embedding rules. To introduce the watermarking scalability, we chose the 3 most significant bits which represent 8 different states corresponding to 6 different symbols (*EZ = Embedded Zero, CZ = Cumulative Zero, WZ = Weak Zero, EO = Embedded One, CO = Cumulative One and WO = Weak One*) to identify the original coefficient's association with a 0 or 1 as shown in Table 1. Now, based on the input watermark stream, if required, new association is made by altering the chosen 3 most significant bits in the tree to reach the nearest symbol, as shown the state machine diagram in Fig. 2. Assuming the current state of the binary tree is EZ, to embed watermark bit 0 no change in state is required while to embed watermark bit 1, a new value of the binary tree must be assigned associated with either WO, CO or EO. However to minimize the distortion, the nearest state change must occur as shown in the state machine diagram. Finally the watermarked image is obtained by dequantizing the modified binary tree followed by the inverse transform.

Symbol	Binary tree	Watermark Association
Embedded Zero (EZ)	000xxxx	0
Embedded Zero (EZ)	001xxxx	0
Cumulative Zero (CZ)	010xxxx	0
Weak One (WO)	011xxxx	1
Weak Zero (WZ)	100xxxx	0
Cumulative One (CO)	101xxxx	1
Embedded One (EO)	110xxxx	1
Embedded One (EO)	111xxxx	1

Table 1: Tree-based watermarking rules table

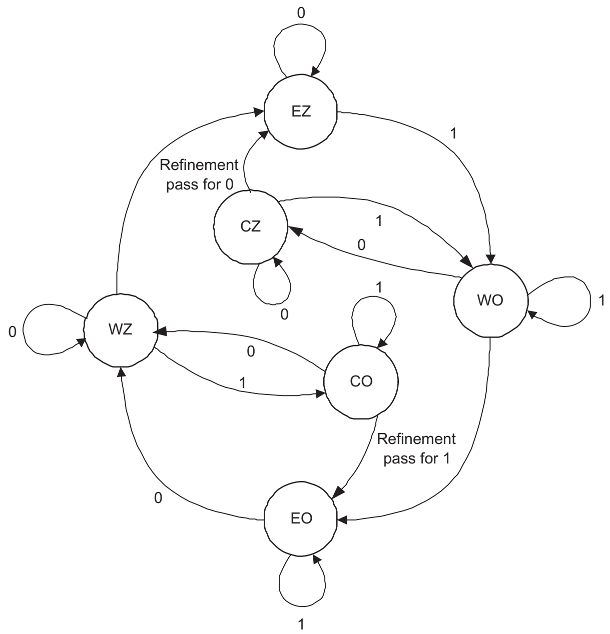

Figure 2: State machine diagram of watermark embedding based on tree symbol association model.

2.1.3 Extraction and Authentication

A universal blind extractor is proposed for watermark extraction and authentication process. The wavelet coefficients are generated after the forward transform on the test

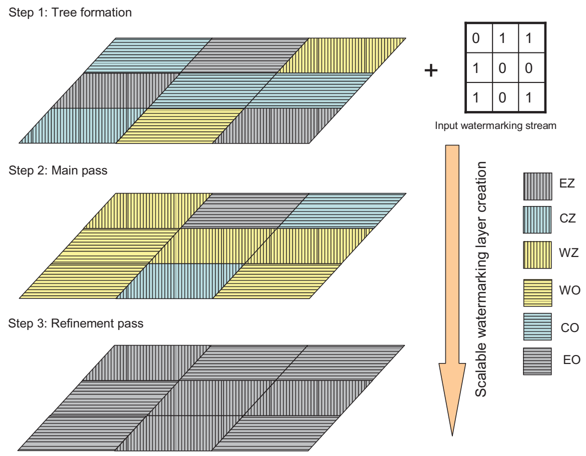

Figure 3: Proposed scalable watermarking layer creation.

image followed by the tree formation process as in embedding. Based on the recovered tree structure, symbols are regenerated to decide on a 0 or 1 watermark extraction. The extracted watermark is then authenticated by comparing with the original watermark by measuring the Hamming distance (Bit Error Rate).

2.2 Incorporating the scalability

In the proposed algorithm, the symbols in Table 1 can be ranked based on the improved associated robustness. The MSB in the binary tree corresponds to a coarse-grained quantization index, whereas LSB represents a fine-grained quantization index. It is evident that to extract the watermark bit successfully, all three MSBs of any binary tree must be unaltered in case of WO, CO or WZ, CZ, whereas only two most significant bit is required to be preserved for EO or EZ. Therefore two consecutive 0s (EZ) or 1s (EO) provides the strongest association with 0 or 1, respectively and thus the robustness. On the other hand, WO, CO and WZ, CZ offer the same level of robustness and hence the robustness rank $\mathcal{R}$ of the symbols can be defined as $\mathcal{R}_{EO} > \mathcal{R}_{CO}, \mathcal{R}_{WO}$ and $\mathcal{R}_{EZ} > \mathcal{R}_{CZ}, \mathcal{R}_{WZ}$. At the same time we can measure the collective embedding distortion rate at scalable atom level by combining distortion and data capacity:

$$\Phi = \frac{\sum_{m=0}^{X-1}\sum_{n=0}^{Y-1}(I(m,n) - I'(m,n))^2}{L}, \qquad (3)$$

where Φ represents embedding distortion rate, I and I' are the original and watermarked image, respectively with dimension $X \times Y$ and L is the number of watermark bits embedded. We exploit these two properties of the algorithm to incorporate the watermarking scalability.

Step 1 (Tree formation): Binary trees are formed using the proposed algorithm for each coefficient and trees are assigned to symbols according to Table 1.

Step 2 (Main pass): In step 2, based on the input watermark stream, we alter the trees to create right association as described in Fig. 2. Hence the coefficients are now rightly associated at least with basic WZ/WO symbol and thus we comfortably name it as the *base layer*. The embedding distortion is calculated progressively at individual tree levels.

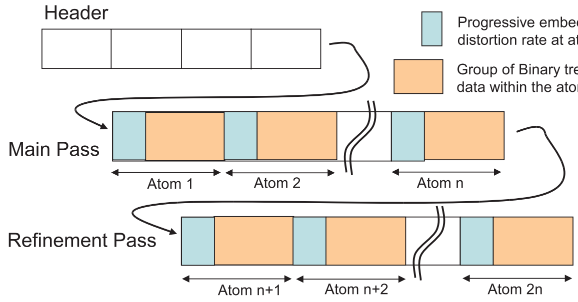

Figure 4: Code stream generation.

Step 3 (Refinement pass): The main aim of the refinement pass is to increase the watermarking strength progressively to increase the robustness. The base layer provides basic minimum association with watermark bits. In the refinement pass, the watermarking strength is increased by modifying the symbols and corresponding tree to the next available level, *i.e.*, $WZ \to EZ, CZ \to EZ, WO \to EO$ and $CO \to EO$ as shown in Fig. 2. At the end of this pass, all trees are modified and associated with the strongest watermark embedding EZ/EO. The distortion is calculated as the refinement levels progress. An example scalable system is shown in Fig. 3.

Step 4 (Hierarchical atom and code stream generation): During two different passes, the binary trees are modified according to the input watermark association and progressive embedding distortion is calculated at each individual tree. Here we define these individual trees or group of trees as an *atom*. Each *atom* contains two pieces of information: embedding distortion rate and the modified binary tree. Now a code stream is generated by concatenating these atoms as shown in Fig. 4. One set of header information is also included at the beginning of the stream to identify the input parameters such as wavelet kernel, decomposition levels, depth of the binary trees etc.

Now, based on the user's requirement, the code stream can be truncated at any desired distortion-robustness atom level and the watermarked image can be generated by dequantizing the truncated code stream, followed by the inverse transform, as described in Section 2.1. Finally the watermark extraction and authentication of the compressed test image is performed using the universal blind extractor as described before.

3 Experimental results and discussion

3.1 Experimental simulations

The experimental simulations are grouped into two sets: 1) Demonstration of the concept and 2) Robustness performance against JPEG 2000. In the experimental sets, a 3 level 9/7 wavelet decomposition is performed using a test data set of 20 images (consists of Kodak test image set and

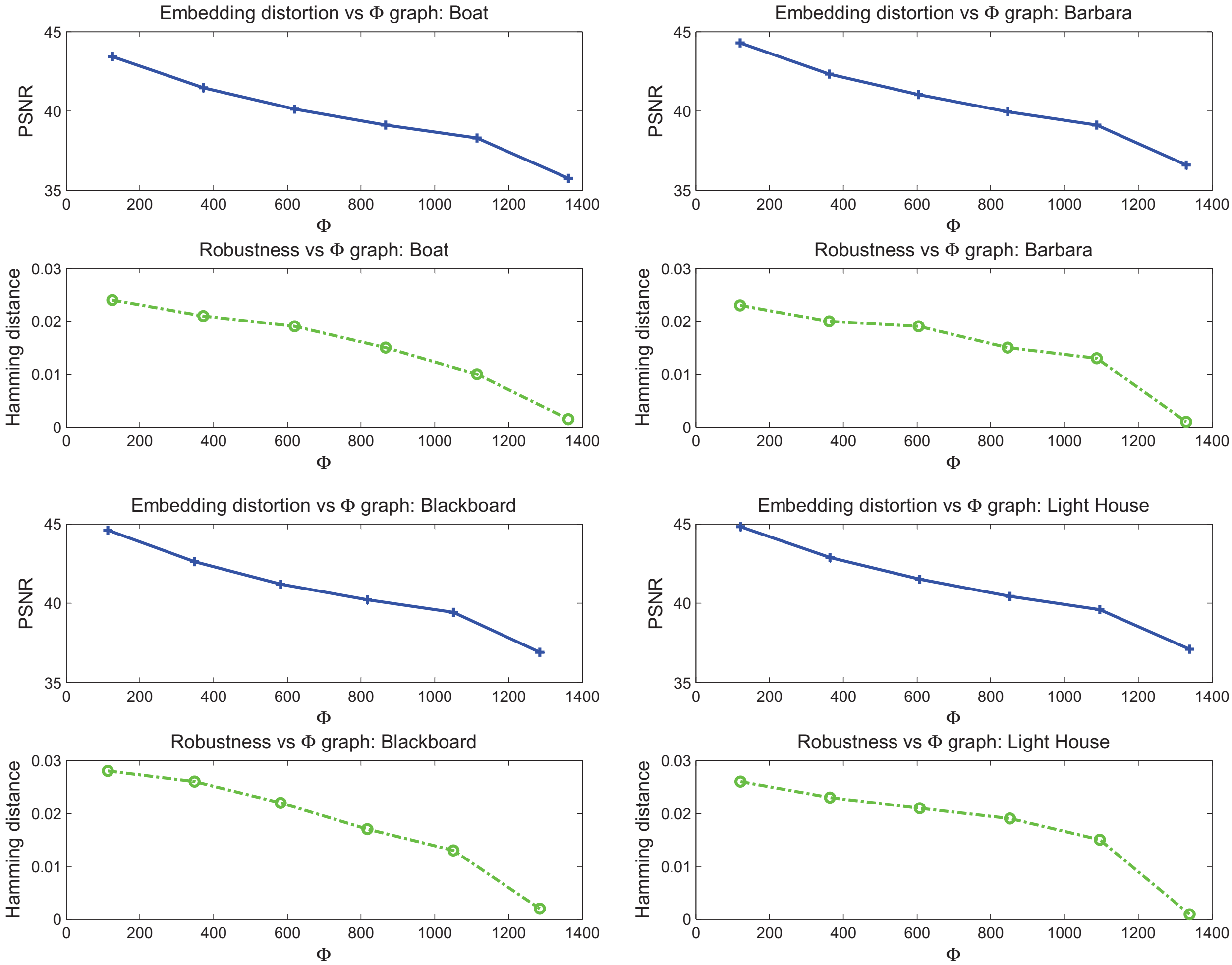

Figure 5: PSNR and robustness vs Φ graph. *Row 1:* Embedding distortion vs. Φ, *Row 2:* Hamming distance vs. Φ.

other popular test images). The low frequency subband has been selected to embed a binary logo based watermark. The initial quantization value, λ, is set to 32, resulting in the tree depth of d=6. In generating the code stream, atoms are defined by grouping every 16 consecutive binary trees.

Demonstration of the concept: Once the code stream is generated, a set of watermarked images is produced by truncating the code stream at different embedding distortion rate points, Φ. As the embedding process creates a hierarchical code stream, at various Φ, the watermark strength varies accordingly, *i.e.,* higher Φ corresponds to higher watermarking strength for a given data capacity. As a result, with increased value of Φ, high embedding distortion is introduced in the watermarked images and hence the image quality degrades. However, with a higher watermarking strength, the robustness improves hierarchically. The overall embedding distortion, measured by PSNR and the robustness (Hamming distance) at various Φ are shown in Fig. 5 for two test images. A lower Hamming distance represents better robustness. The results clearly show that with increasing Φ, *i.e.,* greater watermarking strength results in a poor PSNR but offers better robustness. However a trade off can be made based on the application scenario by selecting an optimum Φ to balance imperceptibility and robustness.

Robustness performance against JPEG 2000: Fig. 6 shows the robustness performance of the proposed watermarking scheme against JPEG 2000 scalable compression and the watermark scalability at a given *depth, d*. It is evident from the plots that higher Φ, for a given depth, offers greater robustness to such scalable content adaptation attacks. The watermark scalability is achieved by truncating the distortion-constrained code stream at various rate points (Φ). With increased Φ, more coefficients are associated with EZ/EO and hence the robustness is improved by successfully retaining the watermark information at higher compression rates. The results shows that more than 60% improvement is achieved between minimum and maximum Φ, at a given d.

3.2 An application scenario of scalable watermarking:

As discussed in Section 1, an experimental application scenario of scalable watermarking is presented here. As an example in Fig. 7, we compared the embedding distortion of the watermarked image after compression. The

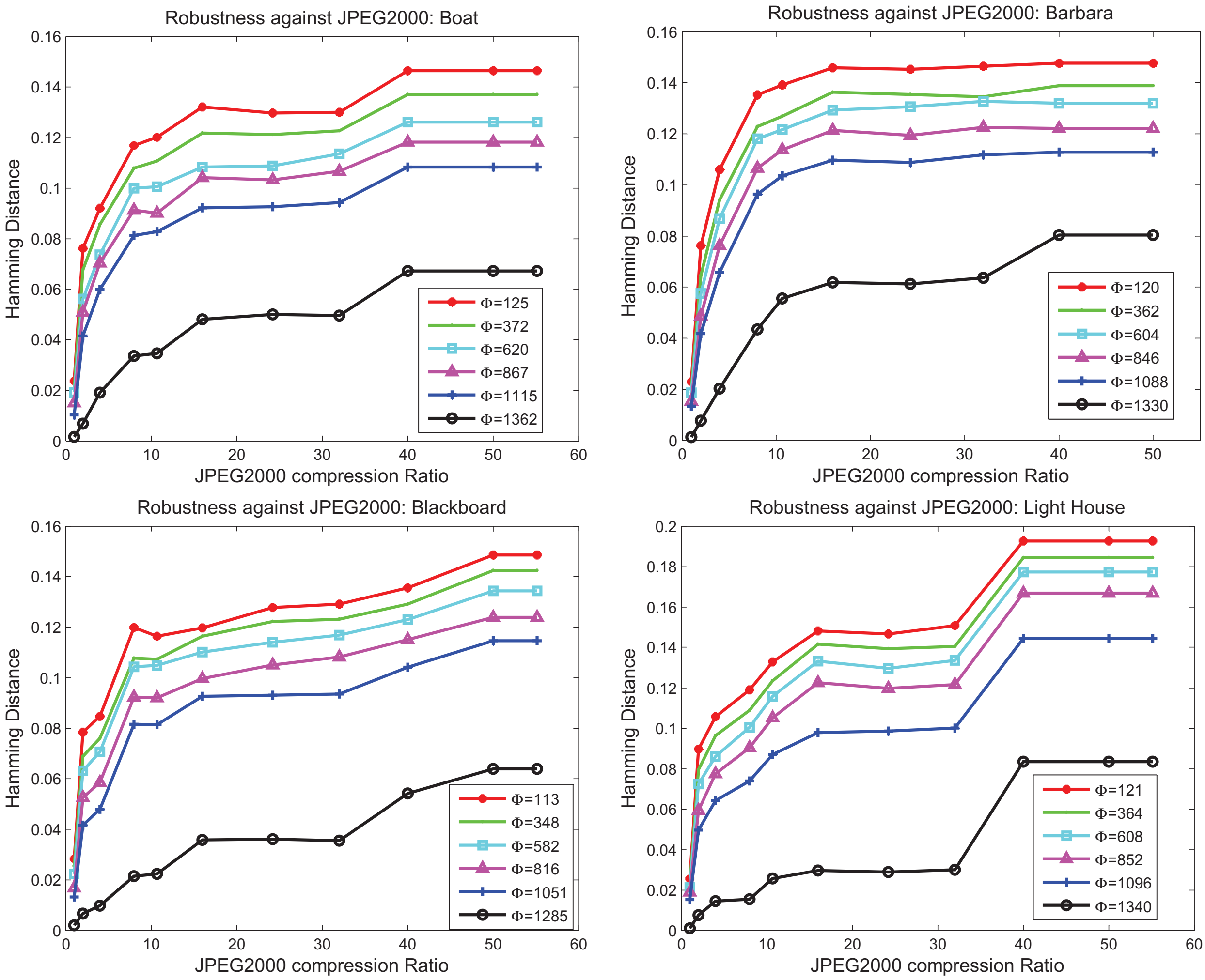

Figure 6: Robustness against JPEG 2000 for various Φ at $d = 6$.

PSNR of the watermarked and the not watermarked images are similar at various compression points, while the watermarked images offer authenticity with desired robustness, *i.e.,* Hamming distance (HD = 0.08). Hence to achieve greater robustness at a high compression ratio (CR), one can choose higher Φ and the effect on embedding distortion is neutralized by compression.

4 Conclusions

In this paper, we proposed a novel concept of scalable watermarking. Firstly a distortion constrained code stream is generated by concatenating hierarchically nested joint distortion robustness coding atoms. The code stream is then truncated at various embedding distortion rate points to create watermarked images, based on the distortion-robustness requirements. The extraction and authentication is done using a blind universal extractor. The concept is experimentally verified for images and the robustness against JPEG 2000 quality scalability is tested.

Acknowledgements

The work was funded by BP-EPSRC Dorothy Hodgkin Post Graduate Award.

References

[1] D. Bhowmik and C. Abhayaratne. The effect of quality scalable image compression on robust watermarking. In *Proc. Int'l Workshop on Digital Signal Processing*, pages 1–8, 2009.

[2] T. P.-C. Chen and T. Chen. Progressive image watermarking. In *Proc. IEEE ICME*, volume 2, pages 1025 –1028, 2000.

[3] P. Meerwald and A. Uhl. Scalability evaluation of blind spread-spectrum image watermarking. In *Proc. Int'l Workshop on Digital Watermarking (IWDW '08), Lect. Notes in Comp. Sci. (LNCS)*, volume 5450, pages 61–75, 2008.

[4] A. Piper, R. Safavi-Naini, and A. Mertins. Resolution and quality scalable spread spectrum image watermark-

Not watermarked: CR=2, PSNR=41.96;

Watermarked: CR=2, PSNR=39.93, Φ=120, HD=0.08;

Not watermarked: CR=50, PSNR=21.61;

Watermarked: CR=50, PSNR=21.47, Φ=1330, HD=0.08;

Figure 7: An application example of scalable watermarking, maintaining embedding distortion and robustness against JPEG 2000.

ing. In *Proc. 7th workshop on Multimedia and Security*, pages 79–90, 2005.

[5] P.-C. Su, H.-J. M. Wang, and C.-C. J. Kuo. An integrated approach to image watermarking and JPEG-2000 compression. *The Journal of VLSI Signal Processing*, 27(1):35–53, Feb. 2001.