

Received October 9, 2018, accepted October 17, 2018, date of publication October 29, 2018, date of current version November 30, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2878456

Adaptive Image Watermarking Algorithm Based on an Efficient Perceptual Mapping Model

TAHA BASHEER TAHA , **RUZELITA NGADIRAN**, AND **PHAKLEN EHKAN**, (Member, IEEE)

School of Computer and Communication Engineering, University Malaysia Perlis, Arau 02600, Malaysia

Corresponding author: Taha Basheer Taha (ph.d.taha@gmail.com)

This work was supported by the Ministry of Education Malaysia through the Fundamental Research Grant Scheme under Grant FRGS/1/2017/ICT05/UNIMAP/02/2.

ABSTRACT Watermark imperceptibility is a significant requirement for keeping watermarked images looking perceptually similar to the original ones. Effective watermark imperceptibility requires the creation of a perceptual model that simulates the human visual system to efficiently hide the watermark in places where the human eye cannot observe it. Current perceptual-based watermarking models use complex computations that are difficult to implement in embedded systems or in real-time applications. In this paper, a low-complexity, integer-based lifting wavelet transform was utilized to create a perceptual mapping model that mainly relies on a new texture mapping model called accumulative lifting difference (ALD). The ALD is combined with a simplified edge detection and luminance masking models to obtain a comprehensive perceptual mapping model that has high-noise tolerance and it is based on low-complexity calculations. The proposed model was 7% faster than the fastest pixel-based compared model with an enhanced average peak signal-to-noise ratio (PSNR) gain of 2.78 dB. In comparison to the largest noise tolerance compared sub-band model, the proposed just noticeable distortion model had a PSNR gain of 1.8 dB and an execution speed that was 90% faster. The perceptual model is utilized in a proposed image watermarking algorithm to determine the maximum watermark embedding intensity that is not visible to the human eye. The experimental results show that the proposed algorithm produced high-quality watermarked images and was robust against different geometric and non-geometric attacks. In addition to its usage in watermarking, the new perceptual model can be used in various image processing and real-time applications.

INDEX TERMS Accumulative lifting differences, image watermarking, lifting wavelet transform, perceptual mapping, texture masking.

I. INTRODUCTION

Copyright protection and authentication of digital images and videos are important current challenges due to the widespread use of digital media throughout the world. Digital watermarking is used to protect intellectual property by embedding a piece of data, such as proprietary information, inside digital media by means of algorithms. A robust copyright protection watermarking algorithm should be able to resist different signal processing operations and attacks that may destroy or remove the watermark. Also, an invisible watermarking system is preferred so that the perceptual quality of the watermarked image is still similar to the original image. However, robustness and invisibility are contrary features of a watermark, because, although more robustness can be obtained with higher intensity watermark embedding, image degradation will be increased. For this reason,

perceptual factors should be studied prior to watermark insertion [1], [2].

The watermark, which is not preferred to be noticed by human eyes, is considered to be an additive noise in image processing perspective. However, the human visual system (HVS) perceives noise differently, depending on the structure of image or the structure of different portions within the same image. Figure 1 shows two identical logos embedded in the sky and on the surface of the mountain in which different visual effects are observed. This is because the mountain's surface has more texture than the sky and hence, the watermark is seen to have less of an impact.

Researchers had analysed such cases and concluded that the HVS can perceive visual alerts that pass a certain threshold called the just noticeable distortion (JND) threshold [3]. Accordingly, design of a JND estimation model while



FIGURE 1. Different perceptual evaluation of the same embedded data.

creating a watermarking system will enhance the watermark invisibility because it finds the places where the watermark can be embedded without being perceived by the human eye. Then, a greater embedding intensity can be achieved under high JND threshold areas, which acquires more robustness while maintaining invisibility. As such, in this paper, a new perceptual mapping model is introduced and implemented to determine the JND threshold before embedding the watermark.

The new model is based on a simplified, LWT-based texture mapping model referred to as Accumulative Lifting Differences (ALD) and is supported by a simplified edge detection technique and luminance masking. The model is employed to determine the maximum embedding intensity within the approximation band of the LWT decomposition where the LWT has been reused for transform domain-based watermark embedding. The embedding method provides the ability to retrieve the watermark using non-informed detectors in which neither the original image nor the original watermark are required for watermark retrieval. The calculations that are involved in the implementation of the proposed algorithm are relies on low-complexity and integer calculations. In the next section (section two), recent related works are reviewed. Section three introduces LWT and its features that can be used in the creation of perceptual maps and in watermark embedding. The new method of perceptual mapping and watermark embedding and extracting processes are explained in section four. Experimental results are presented in section five, including a comparison of the proposed model with recent studies and embedding in coloured images. The paper is concluded in section six.

II. RELATED WORK

Before embedding watermarks, researchers utilize or originate their own JND models to simulate the HVS for the purpose of estimating the maximum distortion that is not visible to the human eye when embedding the watermark. These JND models reflect the amount of texture, luminance masking, contrast, or other HVS characteristics that impact the appearance of watermarked images.

An early JND model, based on discrete cosine transform (DCT), was presented by Watson [4]. In this model,

three factors are involved in the simulation of HVS behaviour, including frequency sensitivity, luminance masking, and contrast masking. The factors are combined to create what is called a Watson distance (D_{wat}) that reflects the amount of perceptual tolerance relative to noise. Although several watermarking algorithms use Watson's model as a perceptual mapping model such as Li *et al.* [5], it is not ideal for watermarking as it does not reasonably provide a maximum-strength transparent watermark and it is not optimum in term of robustness [1], [6]. A DCT-based perceptual watermarking model was presented by Tao and Dickinson [7] where DCT blocks were classified according to three factors: luminance masking, edge masking, and texture masking. DCT was also utilized in the model presented by Niu *et al.* [6] in which the spatial contrast sensitivity function (CSF), luminance adaptation, and contrast masking were combined to produce a JND estimator. The function of the model is to determine the maximum watermark insertion bounds according to the host image. Fazlali *et al.* [8] proposed a watermarking method where two-level Contourlet Transform (CT) on the original image is applied. The first level approximate image is partitioned into blocks and then by concatenating some portions of the second level sub bands blocks are formed. The severity of the embedding is determined by two factors: the edge concentration and entropy of approximation band block, and the embedding is performed on the detail band.

Although acceptable performance is achieved by the DCT in creating perceptual mapping-based watermarking models, the watermarked images suffer from the issue of block-artifacts in many cases. The utilization of the discrete wavelet transform (DWT) in watermarking results in high performance due to the temporal-spatial characteristics of DWT, which are similar to HVS behaviour [2]. Furthermore, if the coefficient is modified in a DWT band, only the corresponding region where the coefficient exists is affected [9]. A DWT-based blind watermarking algorithm was presented by Barni *et al.* [9]. The authors estimated HVS characteristics by analysing the imperceptibility of four DWT decompositions. Three considerations were taken into account while evaluating behaviour of the human eye: the lower sensitivity of the eyes to noise in high-resolution bands and in bands having a 45° orientation, the lower sensitivity of the eyes to bright and dark intensities, and the lower sensitivity of the eyes to textured areas, but, within these areas, the HVS is sensitive to change near the edges. Another wavelet-based study was presented by Akhbari and Ghaemmaghami [10] in which entropy masking was utilized to enhance imperceptibility. As entropy is defined as "the measure of information in a signal," higher entropy indicates more complexity in the image and less sensitivity of HVS. However, the extracting process was performed using a non-blind method in which the existence of the original image was necessary to recover the watermark. In addition, the logarithmic calculations that were used to determine entropy masking are complex operations that increase the computational complexity. The traditional DWT, which mainly relies on the Fourier transform, contains

exponential and imaginary numerical calculations, which increase the complexity of the processing task.

Hybrid transform domain watermarking is an alternative to the use of perceptual mapping for obtaining robust and invisible watermarks, in which the transform domain coefficients that having the required robustness and invisibility within each transformation are selected for embedding. Hybrid transform domain watermarking was used in different studies, including models proposed by Roy and Pal [11] that used Redundant Discrete Wavelet Transform (RDWT) with DCT, Poonam and Arora [12] where DWT was combined with SVD, and Hamidi *et al.* [13] in which a hybrid of DFT, DCT, and Arnold transform was used. However, in these hybrid attempts, computational overhead was high because of the use of multiple transform domain computations [11]. Therefore, there is a need for a simple and high-performing perceptual-based watermarking system that can be utilized in limited resource systems and in real-time applications. All the models presented in this paper can be performed by relying solely on simple adder and shift registers, so it can be easily implemented on FPGA-based devices to enhance the performance of current designs of JND systems or watermarking systems [14]–[16], or it can be used with GPU-based image processing systems [17]–[19].

III. LIFTING WAVELET TRANSFORM

LWT, first introduced by Sweldens [20] is considered to be the second generation of wavelets. The main difference that distinguishes LWT from classic wavelet construction is that Fourier transform is not used in the signal transform calculations. Compared to the conventional structure, LWT has better computational efficiency in terms of using a lower number of mathematical operations, which results in a smaller implementation area, less power consumption, and lower design complexity. Due to its “in place calculation” where the produced coefficients replaces the old image pixels without extra memory utilization, and its integer-to-integer transformation abilities, LWT can be easily implemented by hardware [21]. The LWT implementation consists of three main steps: split, predict, and update. A brief explanation of each is presented below.

A. SPLIT

Assume that Y is a dataset that is to be explored to obtain a more compact presentation of the data. In the split step, the data is split into two subsets (akin to cutting an image into two parts with scissors) that are called wavelet subsets. However, the split must be made in such a way that the original Y can be restored from the two subsets. To find a way to split the data, consider Y as an image. It would not be efficient to split Y into two equal parts from the middle, as this would make it difficult to discern the contents of the right side of the image from its left side. Hence, it is preferred to split the image into even and odd samples such that a relationship still exists between the two split samples, which makes it possible

to predict the value of one sample based on the associated split sample [22].

B. PREDICT

A pixel value in an odd position (X_o) is predicted by the values of its two neighbours at even positions (X_e). The difference between the predicted value of a pixel in an odd position and its actual value is stored in the location of odd samples. The signal after the prediction step in odd locations corresponds to the detail band, D_n (Equation 1).

$$D_n = X_o - \text{PREDICT}(X_e) \quad (1)$$

In the gradient areas where pixels intensities changes linearly, the coefficients' predicted values are close to the real values. As a result, the values of the D_n coefficients are near zero. However, in textured areas with high variations in pixels intensities, the detail band coefficients have higher values; i.e., higher coefficient values in the detail band D_n coefficients of a certain area indicates higher divergence in the pattern within that area. This feature is exploited in the creation of the proposed texture masking model.

C. UPDATE

Calculation of the average signal of two samples produces a decomposed signal with half energy [22]. Samples in even positions are considered as the average of their adjacent odd samples. However, due to the non-linearity of the image pixels, even-positioned samples need to be updated with the differences computed in the predict step. The generated signal is the approximation band, S_n , which has the same features of the real image but with half the actual size, (Equation 2). The approximation band S_n was used to extract the edges and calculate the luminance masking in the perceptual mapping model.

$$S_n = X_e + \text{UPDATE}(D_n) \quad (2)$$

The inverse of the lifting scheme is performed by reversing the order of operations and exchanging the sign of the predict and update steps [21].

IV. METHODOLOGY

Figure 2 depicts the flowchart of the proposed watermarking algorithm as two phases: perceptual map creating and watermark embedding, where the new proposed equations and methodologies are highlighted in the diagram blocks of the flow chart. The perceptual mapping phase is presented in the first part of this section, which involves the application of LWT to the original image to obtain approximation band ($S1$), two middle frequency bands ($D1$, $S2$) and detail band ($D2$). Since $S1$ band has the original characteristics of the image, it was utilized to create each luminance mask and edge mask. The $D2$ band was employed to create the new texture mask according to the proposed ALD method. The combination of texture, edge and luminance masks produces the final perceptual mask that decides the watermark embedding strength for

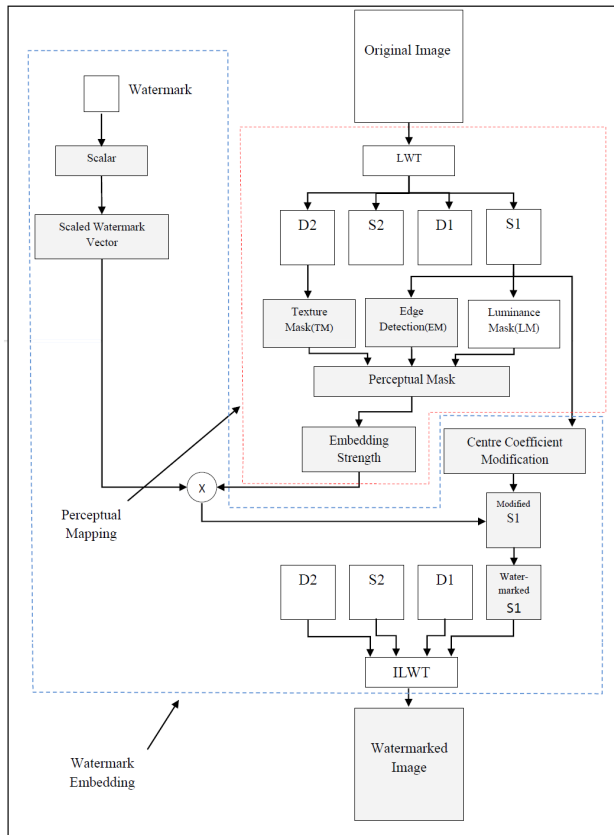


FIGURE 2. Flow diagram of the proposed perceptual based watermarking model.

each 5×5 block according to HVS sensitivity. In the watermarking phase, which is discussed in following subsections, the binary watermark was converted to scaled vector with values $\{-1,1\}$ and embedded within approximation band by multiplying its value by the relative embedding strength that was calculated using the perceptual map. The value is either added to or subtracted from the modified centre coefficient, where the values of the centre coefficients in each block were previously modified to be equal to the average of the surrounding neighbours. Finally, the inverse of LWT (ILWT) was applied on the embedded S1 with the rest of decomposed bands to produce the watermarked image.

Watermark extraction was achieved by comparing the centre coefficient of each internal block with the average of its surroundings. A larger centre coefficient value indicated that the original watermark value was equal to one; otherwise, it was equal to zero. A detailed description of each phase of the proposed method is presented in the following subsections.

A. PERCEPTUAL MAPPING MODEL

The perceptual limitation of the HVS causes the noise below the JND threshold to be imperceptible to the human eye. In the proposed model, the human eye is considered to be less sensitive to noise in the following places [7], [9], [23], [24]:

- 1 - In very dark and very bright intensities.
- 2 - Highly textured areas except for edges.

Accordingly, three models, including luminance masking, texture masking, and edge detection, were used and combined in the new model to simulate the behaviour of the human eye in finding the JND threshold.

1) LUMINANCE MASKING

The HVS is less sensitive to changes in very dark and very bright areas [9], [25]. In the presented model, similar JND weights were assigned to dark areas and white areas [26]. The mathematical equation of luminance adaptation is given by Equation (3):

$$LM(i,j) = \begin{cases} \lfloor (S(i,j) - 128)/16 \rfloor & \text{if } S(i,j) > 127 \\ \lfloor 128 - (S(i,j)/16) \rfloor & \text{Otherwise} \end{cases} \quad (3)$$

where LM is the luminance mask and S (i, j) is the intensity of the approximation band coefficient at the location (i, j).

2) TEXTURE MASKING

Using the statistical approach, the texture concentration in a certain region can be found by calculating the differences of the intensities in that region [27]. Thus, a larger variance in intensity distribution indicates the existence of greater texture. As mentioned in the previous section, the value of the detail band is larger when the difference between the predicted (linear) change and the real change is greater. In other words, the detail band coefficients in a certain region can be considered to be a measure of the non-linearity in that region. Finding the divergence of this non-linearity produces an estimation value for the texture amount in that area.

The detail band was divided into 5×5 blocks, where the adjacent values in each row were subtracted from each other and the absolute values of these differences were accumulated (Figure 3), hence, the proposed model is referred to as Accumulative Lifting Difference (ALD). In ALD, each block owned a value that is correspondent the amount of its own texture. The ALD is given in Equation (4):

$$ALD(I, J) = \sum_{i=l-2}^{l+2} \sum_{j=j-1}^{j+2} |D2(i, j - 1) - D2(i, j)| \quad (4)$$

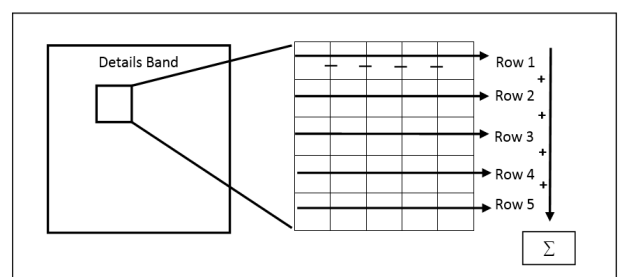


FIGURE 3. ALD Computation.

where I, J are centre coefficients of each 5×5 block in the detail band (D2) and i, j indicate the index of the coefficient in the block.

HVS is sensitive to alterations if the texture is within the area of the edges [9]. The following subsection introduces a simplified edge detection technique for the extraction of edges from the proposed texture mask.

3) EDGE EXTRACTION

The human eye can detect significant changes in the brightness of an image contents at its edges; additionally, the observer may have prior information about how the edge looks. Hence, changes in edge regions are more detectable than other regions [28], [29]. Accordingly, edges need to be extracted from textured areas to create a reliable perceptual map.

Conventional edge detection techniques, such as the Canny edge detection algorithm, involve relatively complex operations such as applying noise removal and smoothing operations [30]. Even simpler edge detectors, such as Sobel or Prewitt methods require a convolution process that involves a task of scanning all the pixels of the image. In the edge detection method proposed in this work, the extended Sobel edge detection kernel published by Kekre and Garge [31] is applied once to each 5×5 block of approximation band instead of moving the kernel over the entire image. By eliminating the convolution step, the number of operations required for applying the four-edge kernels to the same band size is reduced from 61,504 (for a 256×256 approximation band) in the traditional convolution method to 2,704 kernel applications. This optimization reduces the complexity of the algorithm and drastically accelerates execution time, with accepted edge detection results.

The extended Sobel kernels used for the vertical and horizontal edges (as described in [31]) are shown in Figures 4(a) and 4(b). For the diagonal edges, new kernels are designed by rotating the original vertical and horizontal operators by 45° , as shown in Figures 4(c) and 4(d).

The absolute value obtained from the application of each kernel is calculated and the largest value among all the kernels application results is considered as the edge value of that block. Since the LWT has a spatio-frequency localization property [9], the edge values that were extracted from the approximation band blocks are positioned in their alternative ALD texture blocks of detail band; hence, they can be subtracted from the texture mask.

4) POOLING

In the pooling stage, the Final Mask (FM) equation is constructed by subtracting the edge detection mask from the texture mask (ALD) and adding the luminance mask. This can be mathematically described as follows:

$$FM = \alpha TM - \beta EM + \gamma LM \quad (5)$$

where α, β and γ are the weighting factors that are used to decide the intensity of each mask.

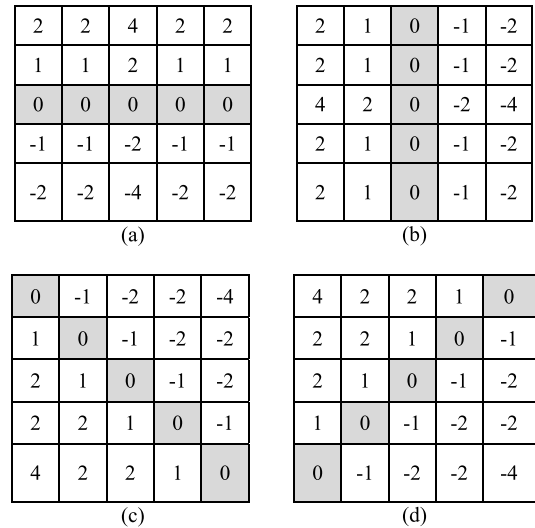


FIGURE 4. Extended Sobel operators.

B. WATERMARKING ALGORITHM

A transform domain-based watermarking algorithm is presented in this section. Blind watermarking algorithms are preferred due to better memory utilization, since it is not necessary to preserve extra memory for original images or original watermarks. Additionally, it is impractical to distribute the original image on watermark detectors in many cases. As such, the proposed embedding process should be designed in a way that allows blind extracting to be performed. The watermarking scheme is divided into watermark embedding and watermark extracting processes, as discussed below.

1) WATERMARK EMBEDDING ALGORITHM

The approximation band was used for watermark embedding for its high robustness. And because of the spatio-frequency feature of the LWT, the perceptual mask in a certain block has the same perceptual value of the alternative block in the approximation band, which is used to control the watermark embedding strength in the approximation band that is sensitive to change. The embedding process is performed according to the following steps:

a: The binary watermark of size $m \times m$ is converted to a one-dimensional vector (wm) after converting each 0 to -1 ($wm\{-1,1\}$). Next, wm is divided into two smaller vectors (wm1 and wm2) in which each part contains a certain number of replications equal to $c1$ and $c2$, respectively, where $c1 + c2 \leq$ embedding capacity. This step adds security to the watermarking algorithm by distributing the watermark all over the bands with variable number of watermark replicas. Also, watermark replication adds extra robustness, since some replicas can compensate for others in case damage is incurred from an attack. The final size of the spread watermark should be:

$$\text{Watermark Vector Size} = \text{size}(wm1) \times c1 + \text{size}(wm2) \times c2 \quad (6)$$

b: LWT decomposition is applied on the original image, and the approximation band produced from the LWT

decomposition is divided into 5×5 blocks, equivalent to the mask blocks of the perceptual map.

c: Each 5×5 block is further divided into nested 3×3 blocks, as shown in Figure 5.

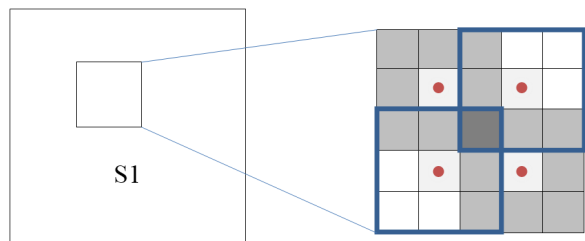


FIGURE 5. Approximation band sub-blocks.

d: The centre coefficient of each 3×3 block is modified to be the average of its eight surrounding coefficients. However, in those blocks where there is a large divergence between maximum and minimum values (difference $>$ threshold, which was experimentally selected to be 64), the value of the centre pixel is taken as the average of its two neighbours having the smaller difference. This exception is made to ensure that changing the centre coefficient to the average of its surrounding coefficients does not affect the visual quality of the image in that block. Also, after finding the average, if the new centre coefficient has a large difference (experimentally greater than 30) relative to the original center coefficient then the block cannot be modified because the modification may affect the visual quality of the image. In this case, the centre coefficient in that block will be returned to its original value. The probability of recurrence for returning the center coefficient to its original value is low because “the brightness level at a point in an image is highly dependent on the brightness levels of neighbouring points unless the image is simply random noise” [32]. Accordingly, the watermark is 100% retrieved in all tested images except for Mandrill image because it contains highly textured areas in some regions that have a center coefficient that cannot be changed.

e: In each 5×5 block, the embedding value, E_m , was calculated as follows:

$$E_m(i, j) = FM(i, j) \times E_s \tag{7}$$

where FM is the final mask value of the perceptual mask in block i, j (Equation 5) and E_s is the embedding strength variable that controls the overall embedding intensity, E_s is experimentally set to be 0.25 in the proposed method.

f: The E_m in each 3×3 block is multiplied by one watermark value and is then added or subtracted from the modified centre coefficient of that block, so that when the watermark value is equal to -1 (which was originally 0), the value is subtracted from the centre coefficient and when the watermark value is equal to 1, the E_m value is added to the centre pixel.

g: ILWT is applied to obtain the watermarked image.

2) WATERMARK EXTRACTING ALGORITHM

The non-informed watermark detector extracts the watermark according to the following steps:

a: A single LWT decomposition is applied to the watermarked image.

b: In each 3×3 block in the approximation band, the maximum difference between the largest and smallest elements is calculated.

c: If the difference is less than the threshold, the average of the eight coefficients in the 3×3 block that surround the center coefficient is calculated and if the difference is greater than the threshold, then the average of the neighbors with the smaller difference is calculated.

d: The average value obtained in step c is subtracted from the center coefficients. The result is a set of positive and negative values.

e: The results related to each watermark bit replicas within blocks that it was originally embedded are added together.

f: If the value obtained from step e was greater than or equal to 0, then the watermark pixel is equal to 1. Otherwise, it is equal to 0.

g: The watermark vector is restored to a two-dimensional binary image.

V. EXPERIMENTAL RESULTS

In the first part of this section, the results of the perceptual model are presented and compared, followed by the results of the watermarking algorithm in terms of invisibility and robustness. The final subsection describes the application of the proposed perceptual mapping and watermarking methods on a colored image.

Two objective metrics were used to evaluate the noise tolerance and visual quality of the tested images: peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM). PSNR is a quantitative metric that determines the quality of the image by comparing the intensity of each pixel in the modified image with its correspondent pixel in the original image. PSNR is calculated from the mean squared error (MSE), which represents the cumulative squared error between two images. The equations for PSNR and MSE are as follows:

$$PSNR = 20 \log_1 0 \frac{255}{\sqrt{MSE}} \tag{8}$$

and

$$MSE = \frac{1}{m * n} + \sum_{i=1}^m \sum_{j=1}^n \|x(i,j) - y(i,j)\|^2 \tag{9}$$

where m, n are the image dimensions and x and y are the original and noisy image, respectively; i, j represent the image pixel index.

A disadvantage of PSNR is that it doesn't consider the different evaluation of the human eye to the same amount of noise in different images structures. SSIM metric provides a better assessment since it takes into consideration three components: luminance, contrast, and structural information, which has better simulation for the human eye observations

and not solely relies on the differences in pixels' intensities [33], [34]. SSIM assigns a specific equation to the three components that are then combined into single equation, as shown in (10) [35]:

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (10)$$

where x is the host image; y is the target image; and μ_x, μ_y , and σ_x, σ_y are the mean intensities and the standard deviations for the images x and y , respectively. C_1 and C_2 are constants.

A. PERCEPTUAL MODEL RESULTS

Figure 6 shows an example of the perceptual masks for three different image according to equation (5), after setting the weighting factors α, β and γ to 0.25, 0.125 and 1, respectively. The brighter areas refer to a higher JND threshold and greater tolerance for noise shaping.



FIGURE 6. Perceptual masks for different images.

The performance of the proposed perceptual model was compared with four different previous models: the fast and low-complexity pixel-based perceptual model [36], a model that based on brain theory (free energy principle) for estimating HVS sensitivity [3], and two other models that consider the pattern orientation method. The first of these used pixel-based domain [37] and the other used DCT sub-band domain [38]. Fifteen standard images (thumbnail images shown in Figure 7, labelled as I1-I15 left to right and top to bottom) are used to compare the models. The Matlab and USC-SIPI image database [39] were used as the resource for the tested images. The codes of the compared models were retrieved from Code Ocean [40].

The comparison was initiated based on two factors: noise shaping, to measure the amount of noise that could be injected



FIGURE 7. Thumbnails of the 15 test images.

within the image, and execution speed that reflects the design complexity.

1) NOISE SHAPING COMPARISON

Generally, a superior JND model should have better perceptual quality for the same amount of additive noise as compared to other models; i.e., it should allow the injection of more noise without being perceived by the human eye [38]. The amount of noise is measured by MSE, while the perceptual quality that simulates human eye perception is measured by SSIM. To compare the proposed model with existing models, noise was injected into the test images according to equation (11) [24], [41]:

$$I_n(i, j) = I(i, j) + n \times r \times FM(i, j) \quad (11)$$

where I is the original image with coordinates i and j , I_n is the image after noise injection, n is a random noise of values $\{1, -1\}$, and r is the regulator factor to control noise energy.

The noise was added with an intensity that corresponded to FM that was produced from the perceptual model. The value of r in Equation (11) was altered to set the SSIM value to 0.9738 [24] for all the images in the compared models. For the same SSIM value, the better model is the one with more noise shaping that is measured by MSE.

The results of the MSE comparisons between the original model and compared models are shown in Figure 8, which shows that the proposed model is able to tolerate more noise in most of the tested images.

2) COMPUTATIONAL COMPLEXITY COMPARISON

In addition to the noise shaping algorithms, the computational speeds of the existing models were compared with the presented model. All five models were executed with MATLAB software on a laptop with CoreI7 CPU 2.2 GHz and 8GB RAM under the same conditions. The execution time for all images for the five models was calculated and is listed in Table 1.

From these results, it is concluded that the proposed JND mapping model is 7% faster than the fastest model which pixel-based model [36] with an enhanced average PSNR gain of 2.78 dB (when calculating the PSNR from the MSE results

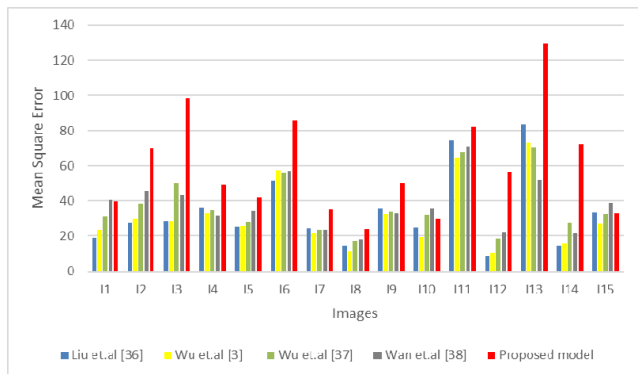


FIGURE 8. Mean Square Error Comparison.

TABLE 1. Execution time for different models.

	Liu et al. [36]	Wu et al. [3]	Wu et al. [37]	Wan et al. [38]	Proposed model
Average Time (Sec)	1.15	4.82	1.52	10.94	1.06

shown in Figure 8) that reflected a greater noise shaping ability. In comparison with the largest noise tolerance model, the sub-band model [38], the proposed model had an execution speed that is 90% faster and a PSNR gain that is 1.8 dB higher. As a result, the presented model has better noise shaping ability and higher execution speed, which make it a promising candidate to be used for watermarking process. The proposed model can be enhanced by proposing a more intensive edge detection technique for the edges that are not detected in the proposed one, this will allow adding more embedding strength without affecting the perceptual quality near the internal and undetected edges.

B. IMAGE WATERMARKING RESULTS

In this section, the perceptual evaluation results for different images are listed in addition to results of the robustness assessment that was conducted while applying various geometric and non-geometric attacks. A comparison of imperceptibility and robustness of the proposed method with a recent hybrid watermarking model is also presented.

1) IMPERCEPTIBILITY EVALUATION

Standard images shown in Figure 7 were used as the host images to embed the watermark. Each image of size 512×512 , so that after one LWT decomposition, each sub-band size will be of size 256×256 . By dividing each subband into blocks of size 5×5 , a set of 51×51 blocks will be created. According to the embedding algorithm, four watermark bits could be embedded in each block; i.e., the embedding capacity is 10,404 bits.

A binary watermark of size 32×32 (Figure 9), which is a 1024-bit vector, was selected for embedding. Hence, the watermark could be distributed according to the spread



FIGURE 9. Binary Watermark.

TABLE 2. PSNR and SSIM for tested watermarked images.

Image	PSNR (dB)	SSIM
11	36.5674	0.9806
12	36.9933	0.9836
13	32.3236	0.9593
14	35.0872	0.9686
15	36.6098	0.9711
16	32.4003	0.9653
17	36.7535	0.9583
18	40.5670	0.9822
19	35.3582	0.9618
110	39.4138	0.9874
111	36.6644	0.9877
112	39.6041	0.9858
113	30.0122	0.9522
114	39.4548	0.9883
115	37.1022	0.9746

spectrum concept with a redundancy of up to 10 times. The objective evaluation, represented by the PSNR and SSIM for the watermarked images, is shown in Table 2 for the watermarked images (which are shown in Figure 10).

From Table 2, lower PSNR values were obtained when more noise was added to the image, which occurred in images that had high noise resistance (as highly textured images). This is due to the fact that the PSNR is a measure of quantity that depends on the MSE, which can be large if the perceptual appearance is still accepted. So, although all obtained PSNR values for the watermarked images were greater than 30 dB, the SSIM provides a better estimation for assessing the quality of the images [24]. SSIM values for all tested images were greater than 0.95, which indicates that the watermarked images have high quality in the objective evaluation and that the watermark is well hidden [42]–[44], in addition to the subjective quality shown in Figure 10.

2) ROBUSTNESS EVALUATION

Different experiments were performed to investigate the functionality of the watermark recovery process and to evaluate the robustness of the watermarking algorithm against



FIGURE 10. Watermarked images.

different geometric and non-geometric attacks. In the first experiment, the watermark was extracted from the watermarked image without attacks, followed by JPEG compression (QF = 50), JPEG compression (QF = 70), Salt and Pepper with a density of 0.02, Gaussian noise (mean = 0 and variance = 0.02), sharpening, LPF, 3 × 3 median filter, 3 × 3 average filter, rotation (0.2), scale (1/2), and cropping of 12.5% from all sides of the image. These attacks were applied to the watermarked image in sequence and then the watermark was extracted and assessed. The similarity between the original and extracted watermarks was computed by the bit error rate (BER) and the normalized correlation coefficient (NCC), according to Equations 12 and 13, respectively.

$$BER = \frac{1}{m \times n} \sum_i^m \sum_j^n [W_{ij} \oplus W'_{ij}] \times 100\% \quad (12)$$

$$NCC = \frac{\sum_i^m \sum_j^n [W_{ij} \oplus W'_{ij}]}{\sqrt{\sum_i^m \sum_j^n (W_{mij})^2} \sqrt{\sum_i^m \sum_j^n (W_{m'ij})^2}} \quad (13)$$

In both equations, W_m, W_m' are the original recovered watermark, each of size $m \times n$.

Results are shown in Table 3, which demonstrate that the watermarked image survived different attacks, based on subjective and objective assessments. The highest robustness produced by the proposed watermarking algorithm was against high pass filtering (sharpening), as the embedding was done in the approximation band. Also, the algorithm had high robustness against additive noise because each watermark bit was distributed as different copies and if the noise affected a certain coefficient, it was covered by the other replicas. Rotation had the worst effect on the extracted watermark because the algorithm works by calculating block average and rotation caused different coefficients to be involved in that calculation. However, rotation of the entire image without cropping the edges and re-rotating it back (as in image viewer applications) did not affect extraction. With cropping, the watermark was distributed throughout the image so that a non-truncated part could be used to retrieve the watermark from the cropped areas. In conclusion, the proposed technique, which relies on LWT approximation band embedding along with a perceptual mask, provided a high perceptual quality images and robust watermarks, and all the calculations were performed using low complexity computations.

3) WATERMARKING RESULTS COMPARISON

The performance of the perceptual adaptive watermarking model proposed in this work was compared with a recent study presented by Kang *et al.* [45], which also aim to find the optimal embedded strength by using hybrid transform method. The comparison is done in terms of imperceptibility and robustness, as discussed below.

a: COMPARISON OF IMPERCEPTIBILITY

Seven standard images (mandrill, lake, pepper, pirate, living room, jet plane and Lena), each 512×512 in size, were used to compare the proposed model and the model of Kang *et al.* SSIM was used as the perceptual metric for the comparison. As shown in Table 4, the proposed model has a higher average SSIM as compared to the average SSIM of the Kang *et al.* model.

b: COMPARISON OF ROBUSTNESS

Results of the robustness comparison are shown in Table 5, In which several attacks were executed on different standard images. The BER average of the recovered watermarks after each attack is shown in Figure 11. For most attacks, the proposed model had a better BER, except in the 5×5 median filter. The reason for this was that the embedding algorithm relied on the 5×5 window and changing the middle value had an effect on all internal 3×3 blocks. However, the images after 5×5 filtering were severely degraded and the watermark has no use if the original image has lost its value [1].

Based on the results, it can be concluded that the proposed model is more robust and generated higher quality watermarked images than the model proposed by Kang *et al.*

TABLE 3. Recovered watermark with BER and NCC values after applying different attacks.

































































































Image	No Attack	JPEG (QF = 50)	JPEG (QF = 70)	Salt and Pepper	Gaussian Noise	Sharpen	LPF	Median Filter	Average Filter	Rotate	Scale Down	Crop	
I1	BER	0	0.0264	0.00098	0.0078	0.0830	0	0.0264	0.0186	0.0264	0.1104	0.0508	0.1875
	NC	1	0.9743	0.9990	0.9923	0.9150	1	0.9747	0.9821	0.9747	0.8880	0.9498	0.8544
	Wm												
I2	BER	0	0.0713	0.0205	0.0137	0.1133	0	0.0352	0.0176	0.0352	0.1201	0.0537	0.1875
	NC	1	0.9326	0.9803	0.9864	0.8845	1	0.9665	0.9829	0.9665	0.8805	0.9468	0.8544
	Wm												
I3	BER	0	0.0352	0.0020	0.0088	0.0723	0	0.0146	0.0117	0.0146	0.0938	0.0195	0.1914
	NC	1	0.9657	0.9981	0.9913	0.9264	1	0.9857	0.9885	0.9857	0.9052	0.9806	0.8520
	Wm												
I4	BER	0	0.0518	0.0137	0.0107	0.0811	0.00098	0.0234	0.0186	0.0234	0.1250	0.0674	0.1787
	NC	1	0.9496	0.9865	0.9894	0.9178	0.9990	0.9771	0.9819	0.9771	0.8728	0.9327	0.8598
	Wm												
I5	BER	0	0.0068	0	0.0049	0.0410	0	0.0059	0.0039	0.0059	0.1221	0.0186	0.1777
	NC	1	0.9932	1	0.9952	0.9590	1	0.9942	0.9961	0.9942	0.8739	0.9816	0.8604
	Wm												
I6	BER	0	0.0039	0.00098	0.0068	0.0166	0	0.0088	0.0088	0.0088	0.1348	0.0273	0.1855
	NC	1	0.9961	0.9990	0.9932	0.9835	1	0.9913	0.9913	0.9913	0.8624	0.9727	0.8556
	Wm												
I7	BER	0	0	0	0.0059	0.0195	0	0.0020	0.00098	0.0020	0.0674	0.00098	0.1758
	NC	1	1	1	0.9942	0.9806	1	0.9981	0.9990	0.9981	0.9323	0.9990	0.8617
	Wm												
I8	BER	0	0.2354	0.1064	0.0166	0.1602	0	0.0137	0.0156	0.0137	0.0752	0.0117	0.1807
	NC	1	0.8244	0.9091	0.9835	0.8344	1	0.9867	0.9849	0.9867	0.9259	0.9885	0.8586
	Wm												

TABLE 3. (Continued.) Recovered watermark with BER and NCC values after applying different attacks.

Image		No Attack	JPEG (QF = 50)	JPEG (QF = 70)	Salt and Pepper	Gaussian Noise	Sharpen	LPF	Median Filter	Average Filter	Rotate	Scale Down	Crop
I9	BER	0	0.00098	0	0.0029	0.0117	0	0.00098	0.0029	0.00098	0.0879	0.0098	0.1709
	NC	1	0.9990	1	0.9971	0.9884	1	0.9990	0.9971	0.9990	0.9104	0.9904	0.8648
	Wm												
I10	BER	0	0.0791	0.0049	0.0186	0.1396	0	0.0176	0.0146	0.0176	0.1025	0.0234	0.1797
	NC	1	0.9277	0.9952	0.9816	0.8541	1	0.9829	0.9859	0.9829	0.8978	0.9768	0.8592
	Wm												
I11	BER	0	0.0830	0.0176	0.0234	0.1260	0.0039	0.0664	0.0449	0.0664	0.1553	0.1143	0.1934
	NC	1	0.9184	0.9828	0.9768	0.8710	0.9962	0.9364	0.9565	0.9364	0.8441	0.8853	0.8508
	Wm												
I12	BER	0	0.2725	0.1426	0.0176	0.1689	0	0.0117	0.0156	0.0117	0.0410	0.0127	0.1807
	NC	1	0.8059	0.8831	0.9825	0.8243	1	0.9886	0.9849	0.9886	0.9593	0.9875	0.8586
	Wm												
I13	BER	0.0029	0.0029	0.0020	0.0020	0.0029	0.00098	0.0020	0.0039	0.0020	0.1035	0.0146	0.1934
	NC	0.9971	0.9971	0.9981	0.9981	0.9971	0.9990	0.9981	0.9961	0.9981	0.8939	0.9855	0.8503
	Wm												
I14	BER	0	0.2822	0.1143	0.0195	0.1719	0	0.0205	0.0195	0.0205	0.0840	0.0254	0.1777
	NC	1	0.7798	0.9008	0.9806	0.8191	1	0.9802	0.9813	0.9802	0.9164	0.9751	0.8604
	Wm												
I15	BER	0	0.0088	0.00098	0.0078	0.0469	0	0.0098	0.0098	0.0098	0.1006	0.0176	0.1807
	NC	1	0.9913	0.9990	0.9923	0.9529	1	0.9903	0.9904	0.9903	0.8991	0.9826	0.8586
	Wm												

TABLE 4. SSIM comparison of watermarked images.

Image	Kang et al.	Proposed Model
Mandrill	0.9795	0.9522
Lake	0.9550	0.9501
Peppers	0.9747	0.9699
Pirate	0.9572	0.9806
Living Room	0.9585	0.9797
Jet plane	0.9613	0.9874
Lena	0.9720	0.9814
Average	0.9655	0.9716

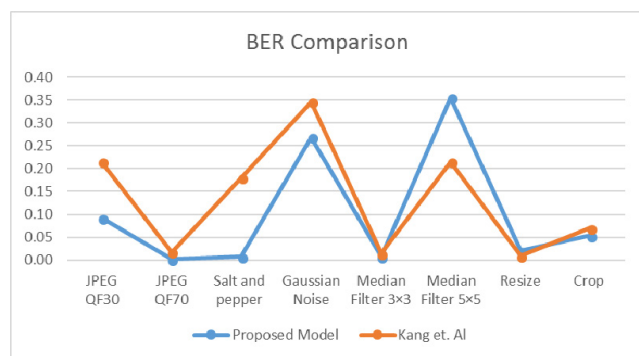


FIGURE 11. BER average for the extracted watermark.

These results were obtained despite the large difference in complexity between the two models, because the model proposed by Kang *et al.* [11] utilized a hybrid form of the DWT, DCT, and SVD to achieve its results that involves high computational complexity. In contrast, the proposed model utilized only the wavelet transform in which a simplified integer LWT was used, which is simpler and faster than the traditional DWT [20].

4) EMBEDDING IN COLORED IMAGES

The application of the proposed watermarking method to RGB (red, green, and blue) colored images was straightforward. The colored images were converted into a YCbCr (luminance, chroma: blue, chroma: red) representation, and the brightness component, Y, was then watermarked in the same way as a grayscale image. After embedding, the watermarked image was converted back to RGB format and displayed. Figure 12 shows the original colored image (a), the watermark (b), the perceptual map (c), and the watermarked image (d). The obtained SSIM for the watermarked image is 0.971 and PSNR is 34.16.

For the extraction, the same process of watermarking is repeated by converting the image into YCbCr and the

TABLE 5. BER comparison of retrieved watermark.

Image	Model	JPEG QF30	JPEG QF 70	Salt and pepper 0.02	Gaussian noise 0.02	Median filter 3x3	Median filter 5x5	Resize (512→256→512)	Crop 25% top left
		Pre-sented	0.0088	0.0020	0.0039	0.1309	0.0039	0.3604	0.0146
Mandrill	Kang Et.al	0.0078	0.0010	0.1035	0.1982	0.0068	0.1436	0.0020	0.0693
	Pre-sented	0.0059	0	0.001	0.1797	0	0.2959	0.0078	0.0820
Lake	Kang Et.al	0.0996	0.0322	0.1621	0.2598	0.0215	0.1064	0.0225	0.0742
	Pre-sented	0.0449	0	0.0020	0.2695	0	0.3213	0	0.0293
Peppers	Kang Et.al	0.977	0.0049	0.2012	0.7787	0.0342	0.1367	0.0020	0.0645
	Pre-sented	0.1074	0.0010	0.0059	0.3213	0.0186	0.3682	0.0508	0.0283
Pirate	Kang Et.al	0.0537	0.0127	0.1924	0.3037	0.0127	0.1357	0.0166	0.0742
	Pre-sented	0.0908	0.0020	0.0098	0.3145	0.0078	0.3525	0.0303	0.0693
Living Room	Kang Et.al	0.0508	0.0068	0.1465	0.2783	0.0176	0.1504	0.0186	0.0693
	Pre-sented	0.2285	0.0049	0.0117	0.3477	0.0146	0.4180	0.0234	0.0244
Jet plane	Kang Et.al	0.1572	0.0439	0.2334	0.2861	0.0020	0.0791	0.0010	0.0693
	Pre-sented	0.1582	0.0020	0.0127	0.3174	0.0039	0.3730	0.0117	0.0254
Lena	Kang Et.al	0.1582	0.0205	0.2217	0.3223	0.0049	0.752	0.0020	0.0664

watermark is then extracted from the Y component. Extraction from the colored image had the same quality as the grayscale images as the same extracting concepts was applied. Figure (13a) shows the extracted watermark from the unaltered watermarked image, the watermark was successfully retrieved with $NCC = 1$ and $BER = 0$. Figure (13b) shows the watermarked image after cropping a large portion. The watermark was mostly retrieved after the cropping as shown in Figure (13c) with $NCC = 0.94$ and $BER = 0.14$.

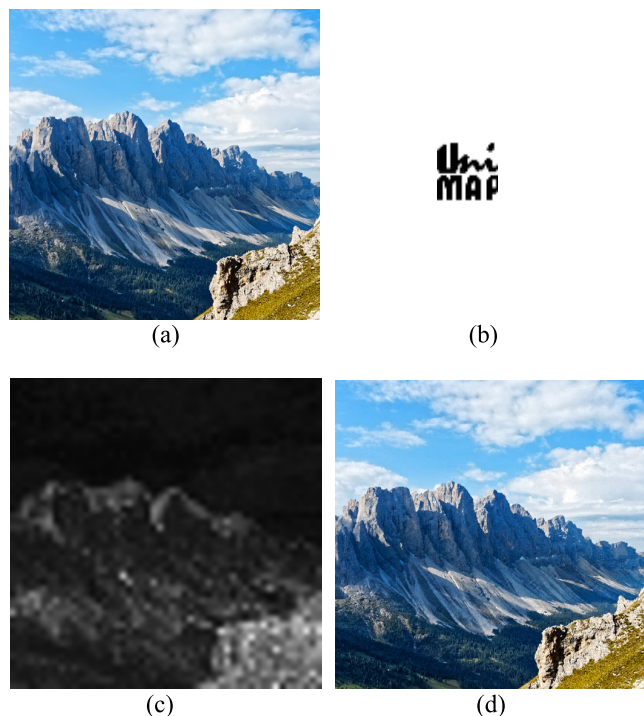


FIGURE 12. Watermark embedding in colored image. a: Original image b:Watermark c:Perceptual map d: Watermarked image.

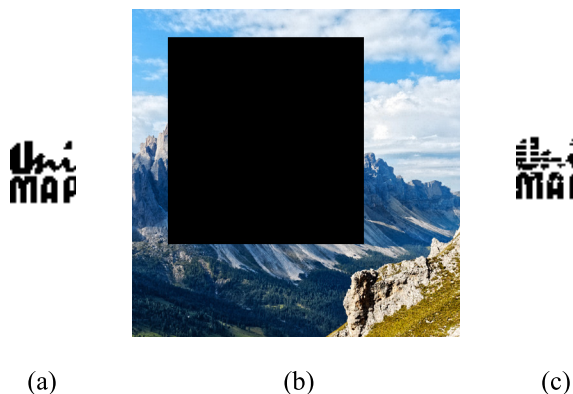


FIGURE 13. Watermark extraction from colored image a: Extracted watermark (no attack), b: Cropped image c: Extracted watermark from the cropped image.

VI. CONCLUSION

LWT, a low complexity, fast, and integer-based signal transformation tool, was utilized for creating a low complexity but highly accurate perceptual mapping model. The detail band of LWT were exploited to create an efficient texture estimation model referred to as ALD model. ALD was combined with a simplified edge detection model and luminance masking to generate an efficient perceptual mapping model that has an execution speed up to 90% faster than recently proposed transform domain model, with noise gain higher by 1.8 dB. The proposed perceptual model was utilized in a new, blind transform domain-based watermarking algorithm to embed the watermark with maximum intensity without

degrading the perceptual quality of the watermarked images. In addition to the high quality of the watermarked images, the watermarking system shows high robustness by resisting different geometric and non-geometric attacks. All the calculations that are involved creating the proposed perceptual and watermarking models can be performed using simple adders and shift registers. The low complexity perceptual model can be used in different signal processing applications and real time systems in addition to its usage in watermarking.

REFERENCES

- [1] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*. San Mateo, CA, USA: Morgan Kaufmann, 2007.
- [2] W. Li, Y. Zhang, and C. Yang, "A survey of JND models in digital image watermarking," in *Proc. Int. Conf. Inf. Technol. Softw. Eng.* Berlin, Germany: Springer, 2013, pp. 765–774.
- [3] J. Wu, G. Shi, W. Lin, A. Liu, and F. Qi, "Just noticeable difference estimation for images with free-energy principle," *IEEE Trans. Multimedia*, vol. 15, no. 7, pp. 1705–1710, Nov. 2013, doi: 10.1109/TMM.2013.2268053.
- [4] A. B. Watson, "DCTune: A technique for visual optimization of DCT quantization matrices for individual images," in *SID Int. Symp. Dig. Tech. Papers*, 1993, pp. 946–949, doi: 10.1109/DCC.1993.253132.
- [5] Y. Li, H. Zhu, R. Yu, G. Yang, and J. Xu, "An adaptive blind watermarking algorithm based on DCT and modified Watson's visual model," in *Proc. Int. Symp. Electron. Commerce Secur.*, Aug. 2008, pp. 904–907.
- [6] Y. Niu, M. Kyan, S. Krishnan, and Q. Zhang, "A combined just noticeable distortion model-guided image watermarking," *Signal, Image Video Process.*, vol. 5, no. 4, pp. 517–526, 2011, doi: 10.1007/s11760-010-0164-x.
- [7] B. Tao and B. Dickinson, "Adaptive watermarking in the DCT domain," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, vol. 4, Apr. 1997, pp. 2985–2988.
- [8] H. R. Fazlali, S. Samavi, N. Karimi, and S. Shirani, "Adaptive blind image watermarking using edge pixel concentration," *Multimedia Tools Appl.*, vol. 76, no. 2, pp. 3105–3120, 2016, doi: 10.1007/s11042-015-3200-6.
- [9] M. Barni, F. Bartolini, and A. Piva, "Improved wavelet-based watermarking through pixel-wise masking," *IEEE Trans. Image Process.*, vol. 10, no. 5, pp. 783–791, May 2001, doi: 10.1109/83.918570.
- [10] B. Akhbari and S. Ghaemmaghami, "Watermarking of still images in wavelet domain based on entropy masking model," in *Proc. IEEE Region 10 TENCON*, Nov. 2005, pp. 1–6.
- [11] S. Roy and A. K. Pal, "A robust blind hybrid image watermarking scheme in RDWT-DCT domain using Arnold scrambling," *Multimedia Tools Appl.*, vol. 76, no. 3, pp. 3577–3616, 2017, doi: 10.1007/s11042-016-3902-4.
- [12] Poonam and S. M. Arora, "A DWT-SVD based robust digital watermarking for digital images," *Procedia Comput. Sci.*, vol. 132, pp. 1441–1448, Apr. 2018, doi: 10.1016/j.procs.2018.05.076.
- [13] M. Hamidi, M. El Haziti, H. Cherifi, and M. El Hassouni, "Hybrid blind robust image watermarking technique based on DFT-DCT and Arnold transform," *Multimedia Tools Appl.*, vol. 77, no. 20, pp. 27181–27214, 2018, doi: 10.1007/s11042-018-5913-9.
- [14] S. Ghosh, P. Ray, S. P. Maity, and H. Rahaman, "Spread spectrum image watermarking with digital design," in *Proc. IEEE Int. Advance Comput. Conf. (IACC)*, Mar. 2009, pp. 868–873.
- [15] S. Liu, Z. Liu, and H. Huang, "FPGA implementation of a fast pipeline architecture for JND computation," in *Proc. 5th IEEE Int. Congr. Image Signal Process. (CISP)*, Oct. 2012, pp. 577–581.
- [16] H. K. Maity and S. P. Maity, "FPGA implementation of reversible watermarking in digital images using reversible contrast mapping," *J. Syst. Softw.*, vol. 96, pp. 93–104, Oct. 2014.
- [17] L. Shi, W. Liu, H. Zhang, Y. Xie, and D. Wang, "A survey of GPU-based medical image computing techniques," *Quant. Imag. Med. Surg.*, vol. 2, no. 3, pp. 188–206, 2012.
- [18] F. Zhu, P. Chen, D. Yang, W. Zhang, H. Chen, and B. Zang, "A GPU-based high-throughput image retrieval algorithm," in *Proc. 5th ACM Annu. Workshop Gen. Purpose Process. Graph. Process. Units*, Mar. 2012, pp. 30–37.

- [19] M. Al-Ayyoub, A. M. Abu-Dalo, Y. Jararweh, M. Jarrah, and M. Al Sa'd, "A GPU-based implementations of the fuzzy C-means algorithms for medical image segmentation," *J. Supercomput.*, vol. 71, no. 8, pp. 3149–3162, 2015.
- [20] W. Sweldens, "Lifting scheme: A new philosophy in biorthogonal wavelet constructions," *Proc. SPIE*, vol. 2569, pp. 68–80, Sep. 1995, doi: [10.1117/12.217619](https://doi.org/10.1117/12.217619).
- [21] M. Gholipour, "Design and implementation of lifting based integer wavelet transform for image compression applications," in *Proc. Int. Conf. Digit. Inf. Commun. Technol. Appl.* Berlin, Germany: Springer, Jun. 2011, pp. 161–172.
- [22] A. Jensen and A. la Cour-Harbo, *Ripples in Mathematics: The Discrete Wavelet Transform*. Berlin, Germany: Springer-Verlag, 2011.
- [23] M. Barni and F. Bartolini, Eds., *Watermarking Systems Engineering: Enabling Digital Assets Security and Other Applications*. Boca Raton, FL, USA: CRC Press, 2004.
- [24] J. Wu, F. Qi, and G. Shi, "An improved model of pixel adaptive just-noticeable difference estimation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2010, pp. 2454–2457.
- [25] C.-H. Chou and Y.-C. Li, "A perceptually tuned subband image coder based on the measure of just-noticeable-distortion profile," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, no. 6, pp. 467–476, Dec. 1995.
- [26] F. Khelifi and K. Zebbiche, "Efficient wavelet-based perceptual watermark masking for robust fingerprint image watermarking," *IET Image Process.*, vol. 8, no. 1, pp. 23–32, 2014, doi: [10.1049/iet-ipr.2013.0055](https://doi.org/10.1049/iet-ipr.2013.0055).
- [27] L. Shapiro and G. Stockman. (Jun. 25, 2018). *Computer Vision*. [Online]. Available: <https://courses.cs.washington.edu/courses/cse576/book/>
- [28] M. P. Eckert and A. P. Bradley, "Perceptual quality metrics applied to still image compression," *Signal Process.*, vol. 70, no. 3, pp. 177–200, 1998.
- [29] H.-C. Lee, C.-R. Dong, and T.-M. Lin, "Digital watermarking based on JND model and QR code features," in *Advances in Intelligent Systems and Applications*, vol. 2. Berlin, Germany: Springer, 2013, pp. 141–148.
- [30] R. Maini and H. Aggarwal, "Study and comparison of various image edge detection techniques," *Int. J. Image Process.*, vol. 3, no. 1, pp. 1–11, 2009.
- [31] H. B. Kekre and S. M. Gharge, "Image segmentation using extended edge operator for mammographic images," *Int. J. Comput. Sci. Eng.*, vol. 2, no. 4, pp. 1086–1091, 2010.
- [32] G. R. Cross and A. K. Jain, "Markov random field texture models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-5, no. 1, pp. 25–39, Jan. 1983.
- [33] Z. Kotevski and P. Mitrevski, "Experimental comparison of PSNR and SSIM metrics for video quality estimation," in *ICT Innovations*. Berlin, Germany: Springer, 2009, pp. 357–366, doi: [10.1007/978-3-642-10781-8_37](https://doi.org/10.1007/978-3-642-10781-8_37).
- [34] K. Silpa and S. ArunaMastani, "Comparison of image quality metrics," *Int. J. Eng. Technol.*, vol. 1, no. 4, Jun. 2012. [Online]. Available: <https://www.ijert.org/phocadownload/V1I4/IJERTV1IS4105.pdf>
- [35] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
- [36] A. Liu, W. Lin, M. Paul, C. Deng, and F. Zhang, "Just noticeable difference for images with decomposition model for separating edge and textured regions," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 11, pp. 1648–1652, Nov. 2010, doi: [10.1109/TCSVT.2010.2087432](https://doi.org/10.1109/TCSVT.2010.2087432).
- [37] J. Wu, L. Li, W. Dong, G. Shi, W. Lin, and C. C. J. Kuo, "Enhanced just noticeable difference model for images with pattern complexity," *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2682–2693, Jun. 2017, doi: [10.1109/TIP.2017.2685682](https://doi.org/10.1109/TIP.2017.2685682).
- [38] W. Wan, J. Wu, X. Xie, and S. G. , "A novel just noticeable difference model via orientation regularity in DCT domain," *IEEE Access*, vol. 5, pp. 22953–22964, 2017.
- [39] (Sep. 15, 2018). *The USC-SIPI Image Database*. [Online]. Available: <http://sipi.usc.edu/database/database.php?volume=misc>
- [40] (2017). *Code Ocean*. Accessed: Jun. 24, 2018. [Online]. Available: <https://codeocean.com>
- [41] M. Uzair and R. D. Dony, "Estimating just-noticeable distortion for images/videos in pixel domain," *IET Image Process.*, vol. 11, no. 8, pp. 559–567, 2017, doi: [10.1049/iet-ipr.2016.1120](https://doi.org/10.1049/iet-ipr.2016.1120).
- [42] T. Zong, Y. Xiang, I. Natgunanathan, W. Zhou, G. Beliakov, and S. Guo, "Robust histogram shape-based method for image watermarking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 5, pp. 717–729, May 2015.
- [43] L.-Y. Hsu and H. T. Hu, "Blind image watermarking via exploitation of inter-block prediction and visibility threshold in DCT domain," *J. Vis. Commun. Image Represent.*, vol. 32, pp. 130–143, Oct. 2015.
- [44] Q. Su, "Novel blind colour image watermarking technique using Hessenberg decomposition," *IET Image Process.*, vol. 10, no. 11, pp. 817–829, 2016.
- [45] X.-B. Kang, F. Zhao, G.-F. Lin, and Y.-J. Chen, "A novel hybrid of DCT and SVD in DWT domain for robust and invisible blind image watermarking with optimal embedding strength," *Multimedia Tools Appl.*, vol. 77, no. 11, pp. 13197–13224, 2017.



TAHA BASHEER TAHA received the B.Sc. degree in computer engineering from the University of Mosul, Mosul, Iraq, and the master's degree in computer engineering from the Institute of Agriculture, Technology and Science, India. He recently completed the Ph.D. degree in computer engineering with University Malaysia Perlis, Perlis, Malaysia. He was appointed as a lecturer at several universities and institutes in Iraq. He was a Graduate Assistant with University Malaysia Perlis. His research interests include digital watermarking, perceptual mapping, image tamper detection, and FPGA design and implementation.



RUZELITA NGADIRAN received the M.Sc. degree in engineering from Malaysia National University, Malaysia, and the Ph.D. degree in engineering of communications and signal processing from Newcastle University, Newcastle upon Tyne, U.K. She is currently a Senior Lecturer with the School of Computer and Communication Engineering, University Malaysia Perlis, Perlis, Malaysia. Her research interests include image compression, signal processing, digital watermarking, and artificial intelligence.



PHAKLEN EHKAN (M'09) received the B.Eng. degree in electrical-electronic engineering from UTM, the M.Sc. degree in information technology from Universiti Utara Malaysia, and the Ph.D. degree in computer engineering from University Malaysia Perlis and the University of Birmingham, U.K.. He was an Engineer/Senior Engineer with MNC- Electronic Industries for six years before joined UniMAP as a Lecturer in 2003, where he is currently an Associate Professor with the School of Computer and Communication Engineering. He is also a Chartered Engineer, U.K., and a Professional Technologist of the MBOT. He has published over 80 articles in international journals and proceedings Scopus indexed. His research interests include reconfigurable computing and FPGA, digital design and embedded systems, digital and image processing, system on chip, smart systems, and IoT. He is a Graduate Member of the BEM and a member of the BCS and IACSIT.

...