**DIGITAL IMAGE WATERMARKING VIA ADAPTIVE LOGO TEXTURIZATION**

**ABSTRACT**

This paper proposes a robust authentication mechanism based on discrete wavelet transform (3-DWT). In this proposed work, a multi-bit watermark (logo) is embedded into the low frequency sub-band of a cover image (input image) by using alpha blending technique. The insertion and extraction of the watermark in the host image/cover image, which is considered as one of the efficient approach than other transform techniques. The embedded watermark (logo) is successfully recovered using Alpha blending extraction and Inverse DWT. We demonstrate that the watermarks generated with the proposed algorithm are invisible and visible and the quality of watermarked image and the recovered image are improved. The image quality of the reconstructed watermark is compared with the existing image watermarking methods by using statistical parameters such as peak-signal-to-noise-ratio (PSNR) and mean square error (MSE).

**CHAPTER 1**

**INTRODUCTION**

**1.1GENERAL**

Digital image processing is the use of computer algorithms to perform image processing on digital images. The 2D continuous image *is* divided into N rows and M columns. The intersection of a row and a column is called a pixel. The image can also be a function other variables including depth, color, and time. An image given in the form of a transparency, slide, photograph or an X-ray is first digitized and stored as a matrix of binary digits in computer memory. This digitized image can then be processed and/or displayed on a high-resolution television monitor. For display, the image is stored in a rapid-access buffer memory, which refreshes the monitor at a rate of 25 frames per second to produce a visually continuous display.

* + 1. **THE IMAGE PROCESSING SYSTEM**

**Digitizer**

**Mass Storage**

**Hard Copy Device**

**Display**

**Image Processor**

**Digital Computer**

**Operator Console**

FIG 1.1 BLOCK DIAGRAM FOR IMAGE PROCESSING SYSTEM

* **Digitizer**

Digitizing or digitizationis the representation of an object, image, sound, document or a signal (usually an analog signal) by a discrete set of its points or samples. Digital information exists as one of two digits, either 0 or 1. These are known as bits.

An image is digitized to convert it to a form which can be stored in a computer's memory or on some form of storage media such as a hard disk or CD-ROM. This digitization procedure can be done by a scanner, or by a video camera connected to a frame grabber board in a computer. Once the image has been digitized, it can be operated upon by various image processing operations.

* Microdensitometer
* Flying spot scanner
* Image dissector
* Videocon camera
* Photosensitive solid- state arrays.
* **Digital computer**

A computer is an electronic device that accepts raw data, processes it according to a set of instructions and required to produce the desired result. Mathematical processing of the digitized image such as convolution, averaging, addition, subtraction, etc. are done by the computer.

* **MASS STORAGE**

Mass storage devices used in desktop and most server computers typically have their data organized in a file system.The secondary storage devices normally used are floppy disks, CD ROMs etc.

* **OPERATOR CONSOLE**

The operator console consists of equipment and arrangements for verification of intermediate results and for alterations in the software as and when require. The operator is also capable of checking for any resulting errors and for the entry of requisite data.

* **Display**

Popular display devices produce spots (display elements) for each pixel:

* Cathode ray tubes (CRTs).
* Liquid crystal displays (LCDs).
* Printers.

Spots may be binary (e.g., monochrome LCD), achromatic (e.g., so-called black-and-white, actually grayscale for intensity), pseudo color or false colors (e.g., for intensity or hyper spectral data), or true color (color data displayed as such).

* + 1. **IMAGE PROCESSING FUNDAMENTAL**

Digital image processing refers processing of the image in digital form. Modern cameras may directly take the image in digital form but generally images are originated in optical form. They are captured by video cameras and digitalized. The digitalization process includes sampling, quantization. Then these images are processed by the five fundamental processes, at least any one of them, not necessarily all of them.

**1.1.2.1 Fundamental steps in image processing**

* 1. Image acquisition
  2. Image preprocessing
  3. Image segmentation
  4. Image representation
  5. Image description
  6. Image recognition
  7. Image interpretation
* **Image acquisition**

First we need to produce a digital image from a paper envelope. This can be done using either a CCD camera, or a scanner

* **Image preprocessing**

This is the step taken before the major image processing task. The problem here is to perform some basic tasks in order to render the resulting image more suitable for the job to follow. In this case it may involve enhancing the contrast, removing noise, or identifying regions likely to contain the postcode.

* **Image segmentation**

Segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as super pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain visual characteristics.

* **Image representation**

Image process is the process of convert the input data to a form suitable for computer processing

* **Image description**

Image description is the process of extract features that result in some quantitative information of interest or features that are basic for differentiating one class of objects from another.

* **Image recognition**

Image recognition is the process of assign a label to an object based on the information provided by its descriptors.

* **Image interpretation**

Image interpretation is the process of assign meaning to an ensemble of recognized objects.

**1.1.2.2 Image types**

There are several ways of encoding the information in an image.

1. Binary image
2. Grayscale image
3. Indexed image
4. True color or RGB image

* **Binary image**

Each pixel is just blackor white. Since there are only two possible values for each pixel (0, 1), we only need one bitper pixel.

* **Grayscale image**

Each pixel is a shade of gray, normally from 0 (black) to 255(white). This range means that each pixel can be represented by eight bits, or exactly one byte. Other grayscale ranges are used, but generally they are a power of 2.

* **Indexed image**

An indexed image consists of an array and a color map matrix. The pixel values in the array are direct indices into a color map. By convention, this documentation uses the variable name X to refer to the array and map to refer to the color map.

* **True Color or RGB image**

Each pixel has a particular color; that color is described by the amount of red, greenand bluein it. If each of these components has a range 0–255, this gives a total of 2563different possible colors. Such an image is a “stack” of three matrices; representing the red, greenand bluevalues for each pixel. This means that for every pixel there correspond 3 values.

**1.1.2.3 image processing goals**

In virtually all image processing applications, however, the goal is to extract information from the image data. Obtaining the information desired may require filtering, transforming, coloring, interactive analysis, or any number of other methods.

To be somewhat more specific, one can generalize most image processing tasks to be characterized by one of the following categories:

Problem Domain

Knowledge Base

Segmentation

Preprocessing

Image Acquisition

Recognition & interpretation

Representation & Description

Result

FIG 1.2 BLOCK DIAGRAM OF FUNDAMENTAL SEQUENCE INVOLVED IN AN IMAGE PROCESSING SYSTEM

1. Image enhancement
2. Image restoration
3. Image analysis
4. Feature extraction
5. Image registration
6. Image compression
7. Image synthesis

* **image enhancement**

This simply means improvement of the image being viewed to the (machine or human) interpreter's visual system. Image enhancement types of operations include contrast adjustment, noise suppression filtering, application of pseudo color, edge enhancement, and many others.

* **image restoration**

The purpose of image restoration is to "compensate for" or "undo" defects which degrade an image. Degradation comes in many forms such as motion blur, noise, and camera misfocus. In cases like motion blur, it is possible to come up with a very good estimate of the actual blurring function and "undo" the blur to restore the original image. In cases where the image is corrupted by noise, the best we may hope to do is to compensate for the degradation it caused.

* **image analysis**

Image analysis is the extraction of meaningful information from images. Image analysis operations produce numerical or graphical information based on characteristics of the original image. They break into objects and then classify them. They depend on the image statistics. Common operations are extraction and description of scene and image features, automated measurements, and object classification. Image analyze are mainly used in machine vision applications.

* **feature extraction**

Feature extraction involves simplifying the amount of resources required to describe a large set of data accurately. When performing analysis of complex data one of the major problems stems from the number of variables involved. Analysis with a large number of variables generally requires a large amount of memory and computation power or a [classification](http://en.wikipedia.org/wiki/Statistical_classification) algorithm which [over fits](http://en.wikipedia.org/wiki/Overfitting) the training sample and generalizes poorly to new samples. Feature extraction is a general term for methods of constructing combinations of the variables to get around these problems while still describing the data with sufficient accuracy.

* **image registration**

Image registration is the process of overlaying two or more images of the same scene taken at different times, from different viewpoints, and/or by different sensors. It geometrically aligns two images the reference and sensed images. The present differences between images are introduced due to different imaging conditions. Image registration is a crucial step in all image analysis tasks in which the final information is gained from the combination of various data sources like in image fusion, change detection, and multichannel image restoration.

Typically, registration is required in remote sensing (multispectral classification, environmental monitoring, change detection, image mosaicing, weather forecasting, creating super-resolution images, integrating information into geographic information systems (GIS)), in medicine (combining computer tomography (CT) and NMR data to obtain more complete information about the patient, monitoring tumor growth, treatment verification, comparison of the patient’s data with anatomical atlases), in cartography (map updating), and in computer vision (target localization, automatic quality control), to name a few.

* **image compression**

The objective of image compression is to reduce irrelevance and redundancy of the image data in order to be able to store or transmit data in an efficient form.

Image compression may be lossy or lossless. Lossless compression is preferred for archival purposes and often for medical imaging, technical drawings, clip art, or comics. This is because lossy compression methods, especially when used at low bit rates, introduce compression artifacts. Lossy methods are especially suitable for natural images such as photographs in applications where minor (sometimes imperceptible) loss of fidelity is acceptable to achieve a substantial reduction in bit rate. The lossy compression that produces imperceptible differences may be called visually lossless.

* **image synthesis**

Image synthesis operations create images from other images or non-image data. Image synthesis operations generally create images that are either physically impossible or impractical to acquire.

**1.1.2**.**4 Applications of image processing**

Image processing has an enormous range of applications; almost every area of science and technology can make use of image processing methods. Here is a short list just to give some indication of the range of image processing applications.

* **Medicine**

Inspection and interpretation of images obtained from X-rays, MRI or CAT scans, analysis of cell images, of chromosome karyotypes. In medical applications, one is concerned with processing of chest X-rays, cineangiograms, projection images of transaxial tomography and other medical images that occur in radiology, nuclear magnetic resonance (NMR) and ultrasonic scanning. These images may be used for patient screening and monitoring or for detection of tumors’ or other disease in patients.

* **Agriculture**

Satellite/aerial views of land, for example to determine how much land is being used for different purposes, or to investigate the suitability of different regions for different crops, inspection of fruit and vegetables distinguishing good and fresh produce from old.

* **DOCUMENT PROCESSING**

It is used in scanning, and transmission for converting paper documents to a digital image form, compressing the image, and storing it on magnetic tape. It is also used in document reading for automatically detecting and recognizing printed characteristics.

* **RADAR IMAGING SYSTEM**

Radar and sonar images are used for detection and recognition of various types of targets or in guidance and maneuvering of aircraft or missile systems.

* **DEFENSE/INTELLIGENCE**

It is used in reconnaissance photo-interpretation for automatic interpretation of earth satellite imagery to look for sensitive targets or military threats and target acquisition and guidance for recognizing and tracking targets in real-time smart-bomb and missile-guidance systems.

**1.2 OBJECTIVE**

Our motivation in this paper is to develop a system to extract the logo from the image from the watermarked image. We are going to separate the original nature of the image and the watermarked separately. In order to implement this task, we are using the watermarking concept.

**1.3 EXISTING SYSTEM**

* The visible watermarking was implemented in early stages.
* Pixel addition based techniques such that the pixel values of the original image and the watermark are combined.
* In existing papers, Watermarking was done in spatial domain. The spatial domain is the normal image space, in which a change in position in image directly projects to a change in position in space. Ex.-Least Significant bit (LSB) method.
  + 1. **DISADVANTAGES OF EXISTING SYSTEM**
* Due to the less robustness of existing watermarking algorithms, the watermarked image looks in different color, so the people can easily identify that some kind of watermark is added.
* Furthermore, image quality may be degraded by the watermark
* The rate–distortion performance is low, and there is a leakage of statistical information.
* The problem in existing scheme is that data is highly sensitive to noise and is easily destroyed.
  + 1. **LITERATURE SURVEY**

**[1] Klimis Ntalianis and Nicolas Tsapatsoulis, “Remote Authentication via Biometrics: A Robust Video-Object Steganographic Mechanism Over Wireless Networks,” in IEEE Transactions on Emerging Topics In Computing, Jan. 2015.**

This paper proposes a robust authentication mechanism based on semantic segmentation, chaotic encryption and data hiding. Assuming that user X wants to be remotely authenticated; initially X’s video object (VO) is automatically segmented, using a head and body detector. Next, one of X’s biometric signals is encrypted by a chaotic cipher. Afterwards the encrypted signal is inserted to the most significant wavelet coefficients of the VO, using its Qualified

Significant Wavelet Trees (QSWTs). QSWTs provide both invisibility and significant resistance against lossy transmission and compression, conditions that are typical in wireless networks. Finally, the Inverse Discrete Wavelet Transform (IDWT) is applied to provide the stego-object (SO).

**DRAWBACK**

* Computational complexity is high.
* There are chances of loss of original data while decrypting, since biometric signals are encrypted by high level transform.
* While Transmission over wireless networks, there may be chances of loss of data.
* DWT algorithm is used, which already exists.

**[2] Mehran Andalibi and Damon M. Chandler,, “Digital Image watermarking via adaptive logo texturization,” in IEEE Transactions on Image processing, Dec. 2015.**

Here, logo is considered as the watermark. The logo should be embedded into the original input image to form the watermarked image using DWT algorithm and the watermark (logo) was decrypted using IDWT. Thus Digital Image watermarking is done.

**DRAWBACK**

* DWT algorithm is used, which already exists.
* Not so robust.

**[3] Huang Lidong, Zhao Wei, Wang Jun and Sun Zebin, “Combination of contrast limited adaptive histogram equalization and discrete wavelet transform for image enhancement,” in IET Transactions on Image processing, 2015, Vol. 9, Issue no. 10, pp. 908–915**

This study presents a novel image enhancement method, named CLAHE-discrete wavelet transform (DWT), which combines the CLAHE with DWT. The new method includes three main steps: First, the original image is decomposed into low-frequency and high-frequency components by DWT. Then, the authors enhance the low-frequency coefficients using CLAHE and keep the high-frequency coefficients unchanged to limit noise enhancement. This is because the high-frequency component corresponds to the detail information and contains most noises of original image. Finally, reconstruct the image by taking inverse DWT of the new coefficients.

**DRAWBACK**

* Watermarking was not done, only image enhancement was done.
* It faces the contrast overstretching and noise enhancement problems.

**[4] Xinpeng Zhang*, Member, IEEE*, Yanli Ren, Liquan Shen, Zhenxing Qian, and Guorui Feng, “Compressing Encrypted Images with Auxiliary information,” IEEE Transactions on Image processing, vol. 16, no. 5, pp. 1327–1336, Aug 2014.**

This paper proposes a novel scheme of compressing encrypted images with auxiliary information. The content owner encrypts the original uncompressed images and also generates some auxiliary information, which will be used for data compression and image reconstruction. At receiver side, the principal image content can be reconstructed using the compressed encrypted data and the secret key.

**DRAWBACK**

* The proposed compression approach is compatible with the modulo-256 addition encryption, but is not suitable for other encryption methods, such as standard stream cipher or AES/DES.
* Low ratio/distortion performance.

**[5] W. Hong and M. Hang, “Robust Digital Watermarking Scheme for Copy Right Protection”, IEEE Trans. Signal Process, vo.l2, pp. 1- 8, 2006.**

W. Hong et al. proposed a robust digital watermarking scheme for copyright protection of digital images based on sub-sampling. The watermark is a binary image, which is embedded in discrete transform coefficient of the host image and not used in the original image. In this scheme, they had used chaotic map in watermarked image. However the result of watermark image is good and robust to attack.

**DRAWBACK**

* Reconstructed image quality is low.
* Watermark is only a binary image.

**[6] Blossom Kaur, Amandeep Kaur, Jasdeep Singh, “Steganographic Approach for hiding Image in DCT Domain”, International Journal of Advances in Engineering & Technology, July 2011.**

Blossom et al. proposed a DCT based watermarking scheme which provides higher resistance to image processing attacks such as JPEG compression, noise, rotation, translation etc. In this approach, the watermark is embedded in the mid frequency band of the DCT blocks carrying low frequency components and the high frequency sub band components remain unused. Watermark is inserted by adjusting the DCT coefficients of the image and by using the private key. Watermark can then be extracted using the same private key without resorting to the original image

**DRAWBACK**

* Not so robust.
* Reconstructed image quality is low.
* DCT is traditional algorithm.

**[7] Baisa L. Gunjal, R.R. Manthalkar, An overview of transform domain robust digital image watermarking algorithms, Journal of Emerging Trends in Computing and Information Sciences, 2010.**

Compared to spatial domain techniques, frequency-domain watermarking techniques proved to be more effective with respect to achieving the imperceptibility and robustness requirements of digital watermarking algorithms. Commonly used frequency-domain transforms include the Discrete Wavelet Transform (DWT), the Discrete Cosine Transform (DCT) and Discrete Fourier Transform (DFT). However, DWT has been used in digital image watermarking more frequently due to its excellent spatial localization and multi-resolution characteristics, which are similar to the theoretical models of the human visual system. Further performance improvements in DWT-based digital image watermarking algorithms could be obtained by increasing the level of DWT.

**[8] X. Xia, C. Boncelet, and G. Arce, A Multiresolution Watermark for Digital Images, Proc. IEEE Int. Conf. on Image Processing, Oct. 1997.**

Xia et.al proposed a watermarking scheme based on the Discrete Wavelet Transform (DWT). The watermark, modeled as Gaussian noise, was added to the middle and high frequency bands of the image. The decoding process involved taking the DWT of a potentially marked image. Sections of the watermark were extracted and correlated with sections of the original watermark. If the cross-correlation was above a threshold, then the watermark was detected. Otherwise, the image was decomposed into finer and finer bands until the entire, extracted watermark was correlated with the entire, original watermark. The problem with the proposed method is that this technique is susceptible to geometric attacks.

**[9] Bhatnagar, G. and Raman, B., A new robust reference watermarking scheme based on DWT-SVD, Elsevier B.V. All rights reserved 2008.**

G. Bhatnagar et al, presented a semi-blind reference watermarking scheme based on discrete wavelet transform (DWT) and singular value decomposition (SVD) for copyright protection and authenticity. Their watermark was a gray scale logo image. For watermark embedding, their algorithm transformed the original image into wavelet domain and a reference sub-image is formed using directive contrast and wavelet coefficients. Then, their algorithm embedded the watermark into reference image by modifying the singular values of reference image using the singular values of the watermark.

**[10] Barni M, Bartolini F, Piva, An Improved Wavelet Based Watermarking Through Pixelwise Masking, IEEE transactions on image processing, 2001.**

M. Barni et al. have developed an improved wavelet-based watermarking through pixel-wise masking. It is based on masking watermark according to characteristics of HVS. The watermark is adaptively added to the largest detail bands. The watermark weighing function is calculated as a simple product of data extracted from HVS model. The watermark is detected by correlation. The proposed method is robust to various attacks but this method is complex than other transform technique.

**[11] D. Kundur and D. Hatzinakos, Digital Watermarking using Multiresolution Wavelet Decomposition, Proceedings, IEEE International Conference Acoustic, Speech, Signal Processing, 1998.**

Kundur et al. decomposed binary logo through DWT. The watermark is scaled by a salience factor, computed on a block by block basis, depending on local image noise sensitivity. It is then repeatedly added to the subbands of DWT decomposition of host image. Visual masking is thus exploited upto only block resolution. A binary code is embedded by suitably quantizing the coefficients of detail bands. For watermark recovery, the embedded binary code is estimated by analyzing coefficients quantization. Once the code is estimated, it is correlated and result is compared to a threshold chosen on the basis of a given false positive probability.

* 1. **PROPOSED SYSTEM**

Here in this work, we are going to detect the hidden watermark (i.e.,logo) from the watermarked image. This concept can be achieved using watermarking techniques. We have implemented a robust image watermarking technique for the logo detection based on Discrete Wavelet Transform (DWT). In this technique a multi-bit watermark (the logo) is embedded into the low frequency sub-band of an input image by using alpha blending technique. Then IDWT is applied to combine alpha blended image with the other sub-bands and high frequency coefficients to form the watermarked biometric image. Here we are going to implement invisible watermarking; hence the logo will be added with the input face image to form the watermarked image. Then at the decryption stage, by using Alpha Blending Extraction Technique we are successfully extracting the watermark content (i.e. logo) present in the watermarked image. We demonstrate that the watermark generated with the proposed algorithm is invisible and the quality of watermarked image and the recovered image are improved. The quality of the extracted image is analyzed by using statistical parameters such as Peak-Signal-to-Noise-Ratio (PSNR) and Mean Square Error (MSE).

In this process we propose an approach of hiding the watermark (logo) into the input image and watermarked image/encrypted image is formed. Then we need to decrypt/extract the watermark and the original image from the encrypted image. First the host image (original input image) is taken as the input and DWT (Discrete Wavelet Transform) is applied to the image which decomposes image into low frequency and high frequency components. In the same manner, DWT is also applied to the watermark image which is to be embedded in the host image. The watermark is embedded into the watermark using alpha blending embedding technique. Then IDWT is applied to combine alpha blended image with the other sub-bands and high frequency coefficients to form the watermarked. The logo (watermark) is recovered from the watermarked image by using alpha blending extraction technique. We demonstrate that the watermarks generated with the proposed algorithm are invisible and visible and the quality of watermarked image and the recovered image are improved. The proposed method is compared with the existing image watermarking methods by using statistical parameters such as peak-signal-to-noise-ratio (PSNR) and mean square error (MSE).

**PROPOSED SYSTEM BLOCK DIAGRAM**

1. **WATERMARK EMBEDDING**

**DWT**

**ORIGINAL INPUT IMAGE**

**WATAERMARKED IMAGE**

**ALPHA BLENDING EMBEDDING TECHNIQUE**

**IDWT**

**DWT**

**WATERMARK (LOGO)**

1. **WATERMARK EXTRACTION**

**WATERMARKED IMAGE**

**IDWT**

**DWT**

**DWT**

**ALPHA BLENDING EXTRACTION TECHNIQUE**

**WATERMARK**

**INPUT FACE IMAGE**

**FIG 1.3: DETAILED BLOCK DIAGRAM OF PROPOSED SYSTEM**

**LL**

**DECOMPOSED IMAGE**

**LOW FREQUENCY COEFFICIENTS**

**DISCRETE WAVELET TRANSFORM (DWT)**

**INPUT FACE IMAGE**

**HH**

**LL**

**HL**

**HIGH FREQUENCY COEFFICIENTS**

**WATERMARKED IMAGE**

**ALPHA BLENDING EMBEDDING TECHNIQUE**

**INVERSE DISCRETE WAVELET TRANSFORM**

**WATERMARK (LOGO)**

**DISCRETE WAVELET TRANSFORM (DWT)**

**FIG 1.4: DETAILED BLOCK DIAGRAM OF PROPOSED SYSTEM**

**1.4.1 PROPOSED TECHNIQUE**

* DISCRETE WAVELET TRANSFORM (DWT) AND INVERSE DWT
* ALPHA BLENDING EMBEDDING TECHNIQUE
* ALPHA BLENDING EXTRACTION TECHNIQUE
  + 1. **PROPOSED SYSTEM ADVANTAGES**
* Output image quality is high.
* It is not sensitive to noise.
* Performance measurements (PSNR) value is high.

**CHAPTER 2**

**PROJECT DESCRIPTION**

**2.1GENERAL**

* **Wavelet Definition**

A ‘wavelet’ is a small wave which has its energy concentrated in time. It has an oscillating wavelike characteristic but also has the ability to allow simultaneous time and frequency analysis and it is a suitable tool for transient, non-stationary or time-varying phenomena.



**Figure 2.1: Representation of a wave (a), and a wavelet (b)**

* **Wavelet Characteristics**

The difference between wave (sinusoids) and wavelet is shown in figure (2.1). Waves are smooth, predictable and everlasting, whereas wavelets are of limited duration, irregular and may be asymmetric. Waves are used as deterministic basis functions in Fourier analysis for the expansion of functions (signals), which are time-invariant, or stationary. The important characteristic of wavelets is that they can serve as deterministic or non-deterministic basis for generation and analysis of the most natural signals to provide better time-frequency representation, which is not possible with waves using conventional Fourier analysis.

* **Wavelet Analysis**

The wavelet analysis procedure is to adopt a wavelet prototype function, called an ‘analyzing wavelet’ or ‘mother wavelet’. Temporal analysis is performed with a contracted, high frequency version of the prototype wavelet, while frequency analysis is performed with a dilated, low frequency version of the same wavelet. Mathematical formulation of signal expansion using wavelets gives Wavelet Transform (WT) pair, which is analogous to the Fourier Transform (FT) pair. Discrete-time and discrete-parameter version of WT is termed as Discrete Wavelet Transform (DWT). DWT can be viewed in a similar framework of Discrete Fourier Transform (DFT) with its efficient implementation through fast filter bank algorithms similar to Fast Fourier Transform (FFT) algorithms.

**2.2 Discrete Wavelet Transform**

The discrete wavelet transform (DWT) is the basic and simplest transform among numerous multiscale transform and other type of wavelet based fusion schemes are usually similar to the DWT fusion scheme.

* **1-D discrete wavelet transform**

The DWT is similar to the DC-CWT except that the input signal is discrete. Therefore, the design rules for ψ(*t*), φ(*t*), *g*[*k*] and *h*[*k*] are similar as in the DC-CWT. The block diagram of the 1-D DWT is illustrated

**Low pass filter**

**g[n]**

**2**

**[n]**

**Input**

**X[n] L-points**

**N-points High pass filter**

**h[n]**

**2**

**[n]**

**L-points**

**The block diagram of the 1-D DWT**

* **Inverse discrete wavelet transform (IDWT)**

The reconstruction process from the DWT coefficients is shown, it is called inverse DWT (IDWT). The filters *h*[*n*], *g*[*n*], *h*1[*n*] and *g*1[*n*] in the figure can be design by quadrature mirror filter (QMF) method or Orthonormal filter method.

**2**

**g[n]**

**2**

**h[n]**

**[n]**

**2**

**X[n]**

**[n]**

**2**

**The block diagrams of DWT and IDWT**

* **2-D discrete wavelet transform**

2-D DWT is very useful for image processing because the image data are discrete and the spatial-spectral resolution is dependent on the frequency. The DWT has the property that the spatial resolution is small in low-frequency bands but large in high-frequency bands. The left-top sub-image (the band with lowest frequencies) has the smallest spatial resolution and represents the approximation information of the original image. Thus, the DWT is suitable for Image compression. On the contrast, the other sub-images (the bands with high frequencies) show the detailed information of the original image. Therefore, these sub-images can be used for edge detection or corner detection.

HH(d)

Restore 2-D image and form 1-D column sequence

2D sequence

HL(v)

Form 1-D row sequence

LH(h)

Restore 2-D image and form 1-D column sequence

LL(a)

**The block diagram of 2-D DWT**

* **Multi-level 2-D discrete wavelet transform**

Multi-resolution analysis is a convenient framework for hierarchical representation of functions or signals on different scales. The basic idea of multiresolution analysis is to represent a function as a limit of successive approximations. Each of these successive approximations is a smoother version of the original function with more and more of the finer “details” added. Wavelets are terminating basis vectors used to decompose signals into a set of coefficients.

The decomposition process can be iterated, with successive approximations being decomposed in turn, so that one signal is broken down into many lower resolution components. This is called the wavelet.



* **Inverse 2-D discrete wavelet transform**

In this case, the reconstruction is lossless, since the filter coefficients are exactly represented and the DWT coefficients are not quantized. However, in practice, we have to transmit the images in limited bandwidth situations, necessitating compression and hence, the DWT coefficients are to be quantized and efficiently encoded

HH(d)

HL(v)

reconstructed image

LH(h)

LL(a)

**The block diagram of inverse 2-D DWT**

**2.3 DISCRETE WAVELET TRANSFORM**

DWT is the multiresolution description of an image the decoding can be processed sequentially from a low resolution to the higher resolution. The DWT splits the image into high and low frequency parts. The high frequency part contains information about the edge components, while the low frequency part is split again into high and low frequency parts. The high frequency components are usually used for watermarking since the human eye is less sensitive to changes in edges.

In two dimensional applications, for each level of decomposition, we first perform the DWT in the vertical direction, followed by the DWT in the horizontal direction. After the first level of decomposition, there are 4 sub-bands: LL1, LH1, HL1, and HH1.

Commonly used frequency-domain transforms include the Discrete Wavelet Transform (DWT), the Discrete Cosine Transform (DCT) and Discrete Fourier Transform (DFT). However, DWT has been used in digital image watermarking more frequently due to its excellent spatial localization and multi-resolution characteristics, which are similar to the theoretical models of the human visual system. Further performance improvements in DWT-based digital image watermarking algorithms could be obtained by increasing the level of DWT.

**2.4 ALPHA BLENDING TECHNIQUE**

Alpha Blending can be accomplished in image processing by blending each pixel from the first source image with the corresponding pixel in the second source image. According to the formula of the alpha blending the watermarked image is given by

WMI=K\*LL4 +Q \* WM2

Where,

WMI = low frequency component of watermarked image

LL4 = low frequency component of the original image obtained by 4-level DWT

WM4 = low frequency component of Watermark image

k, q = Scaling factors for the original image and watermark respectively.

According to the formula of the alpha blending the recovered image is given by

RW = (WMI - k \* LL4)

Where

RW= Low frequency approximation of Recovered watermark,

LL4= Low frequency approximation of the original image

WMI= Low frequency approximation of watermarked image.

**2.5 METHODOLOGIES**

**2.5.1 MODULE NAMES**

* Input Image conversion
* Watermark Embedding
* Watermark Extraction
* Performance measurements

**MODULE 1**

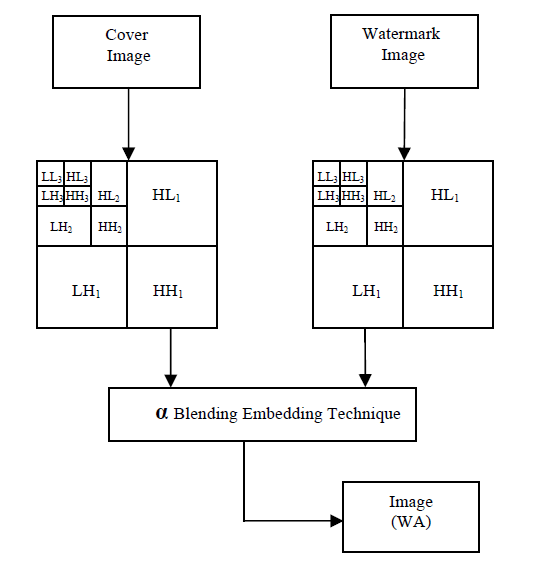
**RGB TO GRAY CONVERSION**

The original image is in an uncompressed format and that the pixel values are within [0, 255], and denote the numbers of rows and columns as N1 and N2 and the pixel number as (N=N1 X N2). Since it was in RGB (color) format, it was converted into grayscale using RGB to gray conversion process. Image Resizing process also done if needed.

**MODULE 2**

**IMAGE ENCRYPTION**

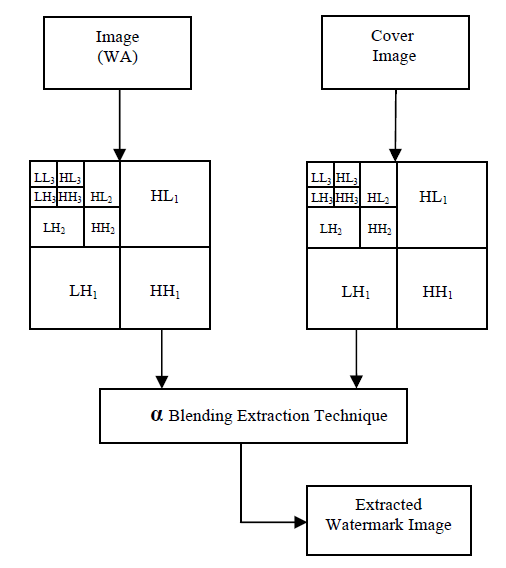
The wavelet transform is applied for the input image and the watermark and then the wavelet decomposed input image and the decomposed logo image is obtained as the output. Then the watermark is embedded into the input image using Alpha-blending embedding technique Then inverse wavelet transform IDWT is applied to combine alpha blended image (mixed low frequency coefficients) with the other sub-bands and high frequency coefficients to form the watermarked image. Finally watermarked/encrypted image (adultered food image) is obtained. This process is called as Encryption/Watermarking.



**MODULE 3**

**IMAGE RECONSTRUCTION/WATERMARK EXTRACTION**

The watermarked/encrypted image (adultered food image) and the original food image is given as the input. Then the watermark (adulterant) is extracted using Alpha blending extraction technique.

****

**Inverse wavelet transform**

By combining the low frequency sub band (LL) with the high frequency subbands by applying IDWT, the output image will contain sharper edges than the watermarked image obtained. This is due to the fact that, the interpolation of isolated high frequency components in high frequency subbands and using the corrections obtained by adding high frequency subbands of DWT of the input image, will preserve more high frequency components after the interpolation than interpolating input image directly.

**MODULE 4**

**PERFORMANCE MEASUREMENTS**

**PSNR**

Peak Signal to Noise Ratio (PSNR) is generally used to analyze quality of image, sound and video files in dB (decibels). PSNR calculation of two images, one original and an altered image, describes how far two images are equal.

**Stationary Wavelet Transform**

The Discrete Wavelet Transform is not a time- invariant transform. The way to restore the translation invariance is to average some slightly different DWT, called decimated DWT, to define the stationary wavelet transform (SWT). Let us recall that the DWT basic computational step is a convolution followed by decimation. The decimation retains even indexed elements. But the decimation could be carried out by choosing odd indexed elements instead of even indexed elements. This choice concerns every step of the decomposition process, so at every level we chose odd or even.

The Stationary wavelet transform (SWT) is a wavelet transform algorithm designed to overcome the lack of translation-invariance of the discrete wavelet transform (DWT). Translation-invariance is achieved by removing the down samplers and up samplers in the DWT and up sampling the filter coefficients by a factor of in the *j* th level of the algorithm. The SWT is an inherently redundant scheme as the output of each level of SWT contains the same number of samples as the input, so for a decomposition of N levels there is a redundancy of N in the wavelet coefficients.

The DWT is not a time-invariant transform. This means that, even with periodic signal extension, the DWT of a translated version of a signal *X* is not, in general, the translated version of the DWT of *X*.

How to restore the translation invariance, which is a desirable property lost by the classical DWT? The idea is to average some slightly different DWT, called ε-decimated DWT, to define the stationary wavelet transform (SWT). This property is useful for several applications such as breakdown point's detection.

**CHAPTER 3**

**SOFTWARE SPECIFICATION**

**3.1 general**

**MATLAB** (**mat**rix **lab**oratory) is a numerical computing environment and [fourth-generation programming language](http://en.wikipedia.org/wiki/Fourth-generation_programming_language). Developed by Math Works, MATLAB allows [matrix](http://en.wikipedia.org/wiki/Matrix_(mathematics)) manipulations, plotting of [functions](http://en.wikipedia.org/wiki/Function_(mathematics)) and data, implementation of [algorithms](http://en.wikipedia.org/wiki/Algorithm), creation of [user interfaces](http://en.wikipedia.org/wiki/User_interface), and interfacing with programs written in other languages, including [C](http://en.wikipedia.org/wiki/C_(programming_language)), [C++](http://en.wikipedia.org/wiki/C%2B%2B), [Java](http://en.wikipedia.org/wiki/Java_(programming_language)), and [Fortran](http://en.wikipedia.org/wiki/Fortran).

Although MATLAB is intended primarily for numerical computing, an optional toolbox uses the [MuPAD](http://en.wikipedia.org/wiki/MuPAD" \o "MuPAD) [symbolic engine](http://en.wikipedia.org/wiki/Computer_algebra_system), allowing access to [symbolic computing](http://en.wikipedia.org/wiki/Symbolic_computing) capabilities. An additional package, [Simulink](http://en.wikipedia.org/wiki/Simulink" \o "Simulink), adds graphical multi-domain simulation and [Model-Based Design](http://en.wikipedia.org/wiki/Model_based_design) for [dynamic](http://en.wikipedia.org/wiki/Dynamical_system) and [embedded systems](http://en.wikipedia.org/wiki/Embedded_systems).

In 2004, MATLAB had around one million users across industry and academia. MATLAB users come from various backgrounds of [engineering](http://en.wikipedia.org/wiki/Engineering), [science](http://en.wikipedia.org/wiki/Science), and [economics](http://en.wikipedia.org/wiki/Economics). MATLAB is widely used in academic and research institutions as well as industrial enterprises.

MATLAB was first adopted by researchers and practitioners in [control engineering](http://en.wikipedia.org/wiki/Control_engineering), Little's specialty, but quickly spread to many other domains. It is now also used in education, in particular the teaching of [linear algebra](http://en.wikipedia.org/wiki/Linear_algebra) and [numerical analysis](http://en.wikipedia.org/wiki/Numerical_analysis), and is popular amongst scientists involved in [image processing](http://en.wikipedia.org/wiki/Image_processing). The MATLAB application is built around the MATLAB language. The simplest way to execute MATLAB code is to type it in the Command Window, which is one of the elements of the MATLAB Desktop. When code is entered in the Command Window, MATLAB can be used as an interactive mathematical [shell](http://en.wikipedia.org/wiki/Shell_(computing)). Sequences of commands can be saved in a text file, typically using the MATLAB Editor, as a [script](http://en.wikipedia.org/wiki/Shell_script) or encapsulated into a [function](http://en.wikipedia.org/wiki/Functional_programming), extending the commands available.

MATLAB provides a number of features for documenting and sharing your work. You can integrate your MATLAB code with other languages and applications, and distribute your MATLAB algorithms and applications.

**3.2 features of matlab**

* High-level language for technical computing.
* Development environment for managing code, files, and data.
* Interactive tools for iterative exploration, design, and problem solving.
* Mathematical functions for linear algebra, statistics, Fourier analysis, filtering, optimization, and numerical integration.
* 2-D and 3-D graphics functions for visualizing data.
* Tools for building custom graphical user interfaces.
* Functions for integrating MATLAB based algorithms with external applications and languages, such as C, C++, Fortran, Java™, COM, and Microsoft Excel.

MATLAB is used in vast area, including signal and image processing, communications, control design, [test and measurement](http://www.mathworks.in/applications/t_m), financial modeling and analysis, and computational. Add-on toolboxes (collections of special-purpose MATLAB functions) extend the MATLAB environment to solve particular classes of problems in these application areas.

MATLAB can be used on personal computers and powerful server systems, including the [Cheaha](http://docs.uabgrid.uab.edu/wiki/Cheaha) compute cluster. With the addition of the Parallel Computing Toolbox, the language can be extended with parallel implementations for common computational functions, including for-loop unrolling. Additionally this toolbox supports offloading computationally intensive workloads to [Cheaha](http://docs.uabgrid.uab.edu/wiki/Cheaha) the campus compute cluster. MATLAB is one of a few languages in which each variable is a matrix (broadly construed) and "knows" how big it is. Moreover, the fundamental operators (e.g. addition, multiplication) are programmed to deal with matrices when required. And the MATLAB environment handles much of the bothersome housekeeping that makes all this possible. Since so many of the procedures required for Macro-Investment Analysis involves matrices, MATLAB proves to be an extremely efficient language for both communication and implementation.

**3.2.1 INTERFACING WITH OTHER LANGUAGES**

MATLAB can call functions and subroutines written in the [C programming language](http://en.wikipedia.org/wiki/C_(programming_language)) or [FORTRAN](http://en.wikipedia.org/wiki/Fortran). A wrapper function is created allowing MATLAB data types to be passed and returned. The dynamically loadable object files created by compiling such functions are termed "[MEX-files](http://en.wikipedia.org/wiki/MEX_file)" (for **M**ATLAB **ex**ecutable).

Libraries written in [Java](http://en.wikipedia.org/wiki/Java_(programming_language)), [ActiveX](http://en.wikipedia.org/wiki/ActiveX) or [.NET](http://en.wikipedia.org/wiki/.NET_Framework) can be directly called from MATLAB and many MATLAB libraries (for example [XML](http://en.wikipedia.org/wiki/XML) or [SQL](http://en.wikipedia.org/wiki/SQL) support) are implemented as wrappers around Java or ActiveX libraries. Calling MATLAB from Java is more complicated, but can be done with MATLAB extension, which is sold separately by Math Works, or using an undocumented mechanism called JMI (Java-to-Mat lab Interface), which should not be confused with the unrelated Java that is also called JMI.

As alternatives to the [MuPAD](http://en.wikipedia.org/wiki/MuPAD" \o "MuPAD) based Symbolic Math Toolbox available from Math Works, MATLAB can be connected to [Maple](http://en.wikipedia.org/wiki/Maple_(software)) or [Mathematica](http://en.wikipedia.org/wiki/Mathematica" \o "Mathematica).

Libraries also exist to import and export [MathML](http://en.wikipedia.org/wiki/MathML" \o "MathML).

**Development Environment**

* Startup Accelerator for faster MATLAB startup on Windows, especially on Windows XP, and for network installations.
* [Spreadsheet Import Tool](http://www.mathworks.in/videos/matlab/new-spreadsheet-import-tool-in-r2011b.html?type=shadow) that provides more options for selecting and loading mixed textual and numeric data.
* Readability and navigation improvements to warning and error messages in the MATLAB command window.
* [Automatic variable and function renaming](http://www.mathworks.in/videos/matlab/new-automatic-variable-and-function-renaming-in-r2011b.html?type=shadow) in the MATLAB Editor.

**Developing Algorithms and Applications**

MATLAB provides a high-level language and development tools that let you quickly develop and analyze your algorithms and applications.

**The MATLAB Language**

The MATLAB language supports the vector and matrix operations that are fundamental to engineering and scientific problems. It enables fast development and execution. With the MATLAB language, you can program and develop algorithms faster than with traditional languages because you do not need to perform low-level administrative tasks, such as declaring variables, specifying data types, and allocating memory. In many cases, MATLAB eliminates the need for ‘for’ loops. As a result, one line of MATLAB code can often replace several lines of C or C++ code.

At the same time, MATLAB provides all the features of a traditional programming language, including arithmetic operators, flow control, data structures, data types, [object-oriented programming](http://www.mathworks.in/products/matlab/object_oriented_programming.html) (OOP), and debugging features.

MATLAB lets you execute commands or groups of commands one at a time, without compiling and linking, enabling you to quickly iterate to the optimal solution. For fast execution of heavy matrix and vector computations, MATLAB uses processor-optimized libraries. For general-purpose scalar computations, MATLAB generates machine-code instructions using its JIT (Just-In-Time) compilation technology.

This technology, which is available on most platforms, provides execution speeds that rival those of traditional programming languages.

Development Tools

MATLAB includes development tools that help you implement your algorithm efficiently. These include the following:

**MATLAB Editor**

Provides standard editing and debugging features, such as setting breakpoints and single stepping

**Code Analyzer**

Checks your code for problems and recommends modifications to maximize performance and maintainability

**MATLAB Profiler**

Records the time spent executing each line of code

**Directory Reports**

Scan all the files in a directory and report on code efficiency, file differences, file dependencies, and code coverage

**Designing Graphical User Interfaces**

By using the interactive tool GUIDE (Graphical User Interface Development Environment) to layout, design, and edit user interfaces. GUIDE lets you include list boxes, pull-down menus, push buttons, radio buttons, and sliders, as well as MATLAB plots and Microsoft ActiveX® controls. Alternatively, you can create [GUIs](http://www.mathworks.in/discovery/matlab-gui.html) programmatically using MATLAB functions.

**3.2.2 ANALYZING AND ACCESSING DATA**

MATLAB supports the entire data analysis process, from acquiring data from external devices and databases, through preprocessing, visualization, and numerical analysis, to producing presentation-quality output.

**Data Analysis**

MATLAB provides interactive tools and command-line functions for data analysis operations, including:

* Interpolating and decimating
* Extracting sections of data, scaling, and averaging
* Thresholding and smoothing
* Correlation, Fourier analysis, and filtering
* 1-D peak, valley, and zero finding
* Basic statistics and curve fitting
* Matrix analysis

**Data Access**

MATLAB is an efficient platform for accessing data from files, other applications, databases, and external devices. You can read data from popular file formats, such as Microsoft Excel; ASCII text or binary files; image, sound, and video files; and scientific files, such as HDF and HDF5. Low-level binary file I/O functions let you work with data files in any format. Additional functions let you read data from Web pages and XML.

**Visualizing Data**

All the graphics features that are required to visualize engineering and scientific data are available in MATLAB. These include 2-D and 3-D plotting functions, 3-D volume visualization functions, tools for interactively creating plots, and the ability to export results to all popular graphics formats. You can customize plots by adding multiple axes; changing line colors and markers; adding annotation, Latex equations, and legends; and drawing shapes.

**2-D Plotting**

Visualizing vectors of data with 2-D plotting functions that create:

* Line, area, bar, and pie charts.
* Direction and velocity plots.
* Histograms.
* Polygons and surfaces.
* Scatter/bubble plots.
* Animations.

**3-D Plotting and Volume Visualization**

MATLAB provides functions for visualizing 2-D matrices, 3-D scalar, and 3-D vector data. You can use these functions to visualize and understand large, often complex, multidimensional data. Specifying plot characteristics, such as camera viewing angle, perspective, lighting effect, light source locations, and transparency.

3-D plotting functions include:

* Surface, contour, and mesh.
* Image plots.
* Cone, slice, stream, and isosurface.

**3.2.3 PERFORMING NUMERIC COMPUTATION**

MATLAB contains mathematical, statistical, and engineering functions to support all common engineering and science operations. These functions, developed by experts in mathematics, are the foundation of the MATLAB language. The core math functions use the LAPACK and BLAS linear algebra subroutine libraries and the FFTW Discrete Fourier Transform library. Because these processor-dependent libraries are optimized to the different platforms that MATLAB supports, they execute faster than the equivalent C or C++ code.

MATLAB provides the following types of functions for performing mathematical operations and analyzing data:

* Matrix manipulation and linear algebra.
* Polynomials and interpolation.
* Fourier analysis and filtering.
* Data analysis and statistics.
* Optimization and numerical integration.
* Ordinary differential equations (ODEs).
* Partial differential equations (PDEs).
* Sparse matrix operations.

MATLAB can perform arithmetic on a wide range of data types, including doubles, singles, and integers.

**CHAPTER 4**

**IMPLEMENTATION**

**4.1 GENERAL**

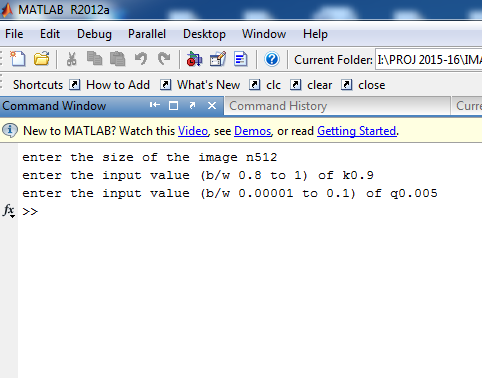
Matlab is a program that was originally designed to simplify the implementation of numerical linear algebra routines. It has since grown into something much bigger, and it is used to implement numerical algorithms for a wide range of applications. The basic language used is very similar to standard linear algebra notation, but there are a few extensions that will likely cause you some problems at first.

**4.2 CODE IMPLEMENTATION**

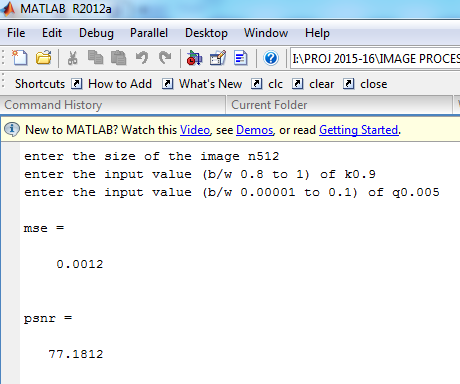
i=(imread('image.png'));

**4.3 SNAPSHOTS**

**INPUT PARAMETERS TO BE GIVEN FOR WATERMARKING PROCESS**

****

**PERFORMANCE MEASUREMENTS**

****

**CHAPTER 5**

**APPLICATIONS**

Our system can be widely used in all areas to overcome the food adulteration process. It detects the adulterant which was mixed or combined with the adulterated image.

* To detect the adulteration in Milk distribution centre.
* To detect the adulteration in hotels
* To detect the adulteration in food product backing industries

**CHAPTER 6**

**CONCLUSION AND REFERENCES**

**6.1 CONCLUSION**

This work proposed an approach of combining or mixing the watermark (logo) into the input image and watermarked image/encrypted image is formed. Then we need to decrypt/extract the adulterant and the original input image from the adulterated food image. We demonstrate that the watermarks generated with the proposed algorithm are invisible and can be visible sometimes and the quality of adulterated image and the recovered image are improved. The proposed method is compared with the existing food adulteration methods by using statistical parameters such as peak-signal-to-noise-ratio (PSNR) and mean square error (MSE).

**6.2 REFERENCES**

[1] L. Yi-bo, X. Hong, and Z. Sen-yue, “The wrinkle generation method for facial reconstruction based on extraction of partition wrinkle line features and fractal interpolation,” in Proc. 4th Int. Conf. Image Graph*.*, Aug. 22–24, 2007, pp. 933–937.

[2] Y. Renner, J. Wei, and C. Ken, “Down sample-based multiple description coding and post-processing of decoding,” in Proc. 27th Chinese Control Conf., Jul. 16–18, 2008, pp. 253–256.

[3] H. Demirel, G. Anbarjafari, and S. Izadpanahi, “Improved motion based localized super resolution technique using discrete wavelet transform for low resolution video enhancement,” in Proc. 17th Eur. Signal Process. Conf., Glasgow, Scotland, Aug. 2009, pp. 1097–1101.

[4] Y. Piao, I. Shin, and H. W. Park, “Image resolution enhancement using inter-subband correlation in wavelet domain,” in Proc. Int. Conf. Image Process., 2007, vol. 1, pp. I-445–448.

[5] H. Demirel and G. Anbarjafari, “Satellite image resolution enhancement using complex wavelet transform,” IEEE Geoscience and Remote Sensing Letter, vol. 7, no. 1, pp. 123–126, Jan. 2010.

[6] C. B. Atkins, C. A. Bouman, and J. P. Allebach, “Optimal image scaling using pixel classification,” in Proc. Int. Conf. Image Process*.*, Oct. 7–10, 2001, vol. 3, pp. 864–867.

[7] W. K. Carey, D. B. Chuang, and S. S. Hemami, “Regularity-preserving image interpolation,” IEEE Trans. Image Process., vol. 8, no. 9, pp. 1295–1297, Sep. 1999.

[8] S. Mallat*,* A Wavelet Tour of Signal Processing, 2nd ed. New York: Academic, 1999.

[9] J. E. Fowler, “The redundant discrete wavelet transform and additive noise,”Mississippi State ERC, Mississippi State University, Tech. Rep. MSSU-COE-ERC-04-04, Mar. 2004.

[10] X. Li and M. T. Orchard, “New edge-directed interpolation,” IEEE Trans. Image Process., vol. 10, no. 10, pp. 1521–1527, Oct. 2001.

[11] K. Kinebuchi, D. D. Muresan, and R. G. Baraniuk, “Waveletbased statistical signal processing using hidden Markov models,” in Proc. Int. Conf. Acoustic., Speech, Signal Process., 2001, vol. 3, pp. 7–11.

[12] S. Zhao, H. Han, and S. Peng, “Wavelet domain HMT-based image super resolution,” in Proc. IEEE Int. Conf. Image Process., Sep. 2003, vol. 2, pp. 933–936.

[13] A. Temizel and T. Vlachos, “Wavelet domain image resolution enhancement using cycle-spinning,” Electron. Lett., vol. 41, no. 3, pp. 119–121, Feb. 3, 2005.

[14] A. Temizel and T. Vlachos, “Image resolution up scaling in the wavelet domain using directional cycle spinning,” J. Electron. Image. vol. 14, no. 4, 2005.

[15] G. Anbarjafari and H. Demirel, “Image super resolution based on interpolation of wavelet domain high frequency subbands and the spatial domain input image,” *ETRI J.*, vol. 32, no. 3, pp. 390–394, Jun. 2010.

[16] A. Temizel, “Image resolution enhancement using wavelet domain hidden Markov tree and coefficient sign estimation,” in Proc. Int. Conf. Image Process*.*, 2007, vol. 5, pp. V-381–384.

[17] J.D. van Ouwerkerk. Image super-resolution survey. Image and Vision Computing, 24:1039–1052, 2006.

[18] E. Vansteenkiste, D.Van der Weken, W. Philips, and E.E. Kerre. Evaluation of the perceptual performance of fuzzy image quality measures. In *proc. KES,* 10th International Conference on Knowledge-Based & Intelligent Information & Engineering Systems, pages 623–630, 2006.

[19] Z. Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: From error visibility to structural similarity. IEEE Transactions on Image Processing, 13:600–612, 2004.