© 2016 IJSRSET | Volume 2 | Issue 2 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

An Efficient Image Watermarking Method using Lifting Wavelet Transformation (LWT)

Keerthana G[*], Bhuvana S, BalaSubramanian R

Department of Computer Science and Engineering, Sri Krishna College of Technology, Coimbatore, TamilNadu, India

ABSTRACT

In image watermarking method we have two common attacks namely, Cropping and random bending. In this paper we propose a method of image-watermarking to deal with these attacks, also with other common attacks. In the embedding process, the pre-processing of host image is done by a Gaussian low-pass filter and then, we select randomly a number of gray levels and the histogram of the filtered image is constructed and these methods are also followed for secret key image . After that, a histogram-shape-related index is given to choose the pixel groups with the highest number of pixels and a safe band is given between the chosen and non-chosen pixel groups. A watermark-embedding scheme is determined to insert watermarks into the chosen pixel groups. The histogram-shape-related index and safe band are used to bring about good robustness. Moreover, a high-frequency component modification mechanism is also applied in the embedding scheme to further improve robustness. At the decoding end, based on the assigned secret key, the watermarked pixel groups are discovered and watermarks are extracted from them.

Keywords: Gaussian filter, Image watermarking, LWT, histogram construction.

I. INTRODUCTION

A digital watermark is a kind of marker covertly embedded in a noise-tolerant signal such as an audio, video or image data. It is typically used to identify ownership of the copyright of such signal. "Watermarking" is the process of hiding digital information in a carrier signal; the hidden information should, but does not need to, contain a relation to the carrier signal. Digital watermarks may be used to verify the authenticity or integrity of the carrier signal or to show the identity of its owners. It is prominently used for tracing copy right infringements and for banknote authentication.

Like traditional watermarks, digital watermarks are only perceptible under certain conditions, i.e. after using some algorithm, and imperceptible otherwise. If a digital watermark distorts the carrier signal in a way that it becomes perceivable, it is of no use. Traditional Watermarks may be applied to visible media (like images or video), whereas in digital watermarking, the signal may be audio, pictures, video, texts or 3D models. A signal may carry several different watermarks at the same time. Unlike metadata that is added to the carrier signal, a digital watermark does not change the size of the carrier signal.

The needed properties of a digital watermark depend on the use case in which it is applied. For marking media files with copyright information, a digital watermark has to be rather robust against modifications that can be applied to the carrier signal. Instead, if integrity has to be ensured, a fragile watermark would be applied.

Both steganography and digital watermarking employ stenographic techniques to embed data covertly in noisy signals. But whereas steganography aims for imperceptibility to human senses, digital watermarking tries to control the robustness as top priority.

Since a digital copy of data is the same as the original, digital watermarking is a passive protection tool. It just marks data, but does not degrade it or control access to the data.

One application of digital watermarking is source tracking. A watermark is embedded into a digital signal at each point of distribution. If a copy of the work is found later, then the watermark may be retrieved from the copy and the source of the distribution is known. This technique reportedly has been used to detect the source of illegally copied movies.

Digital watermarking may be used for a wide range of applications, such as:

- Copyright protection.
- Source tracking (different recipients get differently watermarked content).
- Broadcast monitoring (television news often contains watermarked video from international agencies).
- Video authentication.
- Software crippling on screen casting programs, to encourage users to purchase the full version to remove it.
- A digital watermark is called robust with respect to transformations if the embedded information may be detected reliably from the marked signal, even if degraded by any number of transformations. Typical image degradations are JPEG compression, rotation, cropping, additive noise, and quantization. For video content, temporal modifications and MPEG compression often are added to this list. A digital watermark is called imperceptible if the watermarked content is perceptually equivalent to the original, un-watermarked content. In general, it is easy to create either robust watermarks or imperceptible watermarks, but the creation of both robust and imperceptible watermarks has proven to be quite challenging. Robust imperceptible watermarks have been proposed as a tool for the protection of digital content, for example as an embedded no-copy-allowed flag in professional video content.
- Digital watermarking techniques may be classified in several ways.
- The information to be embedded in a signal is called a digital watermark, although in some contexts the phrase digital watermark means the difference between the watermarked signal and the cover signal. The signal where the watermark is to be embedded is called the host signal. A watermarking system is usually divided into three distinct steps, embedding, attack, and detection. In embedding, an algorithm accepts the host and the data to be embedded, and produces a watermarked signal.
- Then the watermarked digital signal is transmitted or stored, usually transmitted to another person. If this person makes a modification, this is called an attack. While the modification may not be malicious, the term attack arises from copyright protection application, where third parties may attempt to remove the digital watermark through modification. There are many possible modifications, for example, lossy compression of the data (in which resolution is diminished), cropping an image or video, or intentionally adding noise.
- Detection (often called extraction) is an algorithm which is applied to the attacked signal to attempt to extract the watermark from it. If the signal was unmodified during transmission, then the watermark still is present and it may be extracted. In robust digital watermarking applications, the extraction algorithm should be able to produce the watermark correctly, even if the modifications were strong. In fragile digital watermarking, the extraction algorithm should fail if any change is made to the signal.

Cropping

Cropping refers to the removal of the outer parts of an image to improve framing, accentuate subject matter or change aspect ratio. Depending on the application, this may be performed on a physical photograph, artwork or film footage, or achieved digitally using image_editing software. The term is common to the film, broadcasting, photographic, graphic design and printing industries.

In the printing, graphic design and photography industries, cropping refers to removing unwanted areas from a photographic or illustrated image. One of the most basic photo manipulation processes, it is performed in order to remove an unwanted subject or irrelevant detail from a photo, change its aspect ratio, or to improve the overall composition. In telephoto photography, most commonly in bird photography, an image is cropped to magnify the primary subject and further reduce the angle of view when a lens of sufficient focal length to achieve the desired magnification directly is not available. It is considered one of the few editing actions permissible in modern photojournalism along with tonal balance, colour correction and sharpening. A crop made from the top and bottom of a photograph may produce an aspect which mimics the panoramic format (in photography) and the widescreen format in cinematography and broadcasting. Both of these formats are not cropped as such, rather the product of highly specialised optical configuration and camera design.

The aspect ratio of a geometric shape is the ratio of its sizes in different dimensions. For example, the aspect ratio of a rectangle is the ratio of its longer side to its shorter side - the ratio of width to height,[1] when the rectangle is oriented as a "landscape".

The aspect ratio is expressed as two numbers separated by a colon (x:y).The values x and y do not represent actual width and height but, rather, the "relation" between width and height. As an example, 8:5, 16:10 and 1.6:1 are the same aspect ratio.

In objects of more than two dimensions, such as Hyper rectangles, the aspect ratio can still be defined as the ratio of the longest side to the shortest side.

The goal is often to focus the viewer’s attention upon the subject, but the ends and means are ultimately at the discretion of the artist. It is accomplished by manipulating the viewpoint of the image, rather than the object(s) within.

Framing, especially in the photographic arts, is primarily concerned with the position and perspective of the viewer. The position of the observer has tremendous impact on their perception of the main subject, both in terms of aesthetics and in their interpretation of its meaning.

For example, if the viewer was placed very far away from a lone subject in an image, the viewer will gather more information about the subjects’ surroundings and bearing, but very little in terms of his emotions. If the setting was in the middle of flat plain, the viewer might perceive a sense of loneliness or that the subject is lost, because the viewer himself cannot find any visual cues to orient the location of the subject. If some foreground elements are put in front of the viewer, partially obscuring the subject, the viewer would take the position of an unseen observer. Especially if the artist chooses to hint malicious intent, a member of the audience might feel uncomfortable looking through the eyes of a stalker.

Digital watermarking is distinctly different from data hashing. It is the process of altering the original data file, allowing for the subsequent recovery of embedded auxiliary data referred to as a watermark.

A subscriber, with knowledge of the watermark and how it is recovered, can determine (to a certain extent) whether significant changes have occurred within the data file. Depending on the specific method used, recovery of the embedded auxiliary data can be robust to post-processing (such as lossy compression).

If the data file to be retrieved is an image, the provider can embed a watermark for protection purposes. The process allows tolerance to some change, while still maintaining an association with the original image file. Researchers have also developed techniques that embed components of the image within the image. This can help identify portions of the image that may contain unauthorized changes and even help in recovering some of the lost data.

A disadvantage of digital watermarking is that a subscriber cannot significantly alter some files without sacrificing the quality or utility of the data. This can be true of various files including image data, audio data, and computer code.

The organization of this document is as follows. In Section 2 Related Works, this section details about the methods like filtering, histogram construction, numbers of bins and width and also about Discrete Wavelet Transform. In Section 3 Proposed Work, presents about Lifting Wavelet Transform and Embedding Method. Discussed in Section 4 Results and Performance Evaluation details about the results and also PSNR and BET calculations and graphs. In Section 5 Conclusion is given to Concludes this paper.

II. METHODS AND MATERIAL

1. Existing Methods

A. Gaussian filtering

In electronics and signal processing, a Gaussian filter is a filter whose impulse response is a Gaussianu function (or an approximation to it). Gaussian filters have the properties of having no overshoot to a step function input while minimizing the rise and fall time. This behaviour is closely connected to the fact that the Gaussian filter has the minimum possible group delay. It is considered the ideal time domain filter, just as the sinc is the ideal frequency domain filter. These properties are important in areas such as oscilloscopes and digital telecommunication systems.

Mathematically, a Gaussian filter modifies the input signal by convolution with a Gaussian function; this transformation is also known as the Weierstrass transform.

The one-dimensional Gaussian filter has an impulse response given by

$$g(x) = \sqrt{\frac{a}{\pi}} \cdot e^{-a \cdot x^2} \tag{1}$$

and the frequency response is given by the Fourier transform

$$\hat{g}(f) = e^{-\frac{\pi^2 f^2}{a}} \tag{2}$$

with f the ordinary frequency. These equations can also be expressed with the standard deviation as parameter

$$g(x) = \frac{1}{\sqrt{2\pi} \cdot \sigma} \cdot e^{-\frac{x^2}{2\sigma^2}} \tag{3}$$

and the frequency response is given by

$$\hat{g}(f) = e^{-\frac{f^2}{2\sigma_f^2}} \tag{4}$$

By writing a as a function of σ with the two equations for $g(x)$ and as a function of σ_f with the two equations for $\hat{g}(f)$ it can be shown that the product of the standard deviation and the standard deviation in the frequency domain is given by

$$\sigma \cdot \sigma_f = \frac{1}{2\pi} \tag{5}$$

where the standard deviations are expressed in their physical units, e.g. in the case of time and frequency in seconds and Hertz.

In two dimensions, it is the product of two such Gaussians, one per direction:

$$g(x, y) = \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{6}$$

where x is the distance from the origin in the horizontal axis, y is the distance from the origin in the vertical axis, and σ is the standard deviation of the Gaussian distribution.

The Gaussian function is for $x \in (-\infty, \infty)$ and would theoretically require an infinite window length. However, since it decays rapidly, it is often reasonable to truncate the filter window and implement the filter directly for narrow windows, in effect by using a simple rectangular window function. In other cases, the truncation may introduce significant errors. Better results can be achieved by instead using a different window function; see scale space implementation for details.

Filtering involves convolution. The filter function is said to be the kernel of an integral transform. The Gaussian kernel is continuous. Most commonly, the discrete equivalent is the sampled Gaussian kernel that is produced by sampling points from the continuous Gaussian. An alternate method is to use the discrete Gaussian kernel which has superior characteristics for some purposes. Unlike the sampled Gaussian kernel, the discrete Gaussian kernel is the solution to the discrete diffusion equation.

Since the Fourier transform of the Gaussian function yields a Gaussian function, the signal (preferably after being divided into overlapping windowed blocks) can be transformed with a Fast Fourier transform, multiplied with a Gaussian function and transformed back. This is the standard procedure of applying an arbitrary finite impulse response filter, with the only difference that the Fourier transform of the filter window is explicitly known.

Due to the central limit theorem, the Gaussian can be approximated by several runs of a very simple filter such as the moving average. The simple moving average corresponds to convolution with the constant B-spline (a rectangular pulse), and, for example, four iterations of a moving average yields a cubic B-spline as filter window which approximates the Gaussian quite well.

In the discrete case the standard deviations are related by

$$\sigma \cdot \sigma_f = \frac{N}{2\pi} \tag{7}$$

where the standard deviations are expressed in number of samples and N is the total number of samples. Borrowing the terms from statistics, the standard deviation of a filter can be interpreted as a measure of its size. The cut-off frequency of a Gaussian filter might be defined by the standard deviation in the frequency domain yielding

$$f_c = \sigma_f = \frac{1}{2\pi\sigma} \tag{8}$$

where all quantities are expressed in their physical units. If σ is measured in samples the cut-off frequency (in physical units) can be calculated with

$$f_c = \frac{F_s}{2\pi\sigma} \tag{9}$$

where F_s is the sample rate. The response value of the Gaussian filter at this cut-off frequency equals exp(-0.5)≈0.607.

However, it is more common to define the cut-off frequency as the half power point: where the filter response is reduced to 0.5 (-3 dB) in the power spectrum, or 1/√2 ≈ 0.707 in the amplitude spectrum (see e.g. Butterworth filter). For an arbitrary cut-off value 1/c for the response of the filter the cut-off frequency is given by

$$f_c = \sqrt{2\ln(c)} \cdot \sigma_f \tag{10}$$

For c=2 the constant before the standard deviation in the frequency domain in the last equation equals approximately 1.1774, which is half the Full Width at Half Maximum (FWHM) (see Gaussian function). For c=√2 this constant equals approximately 0.8326. These values are quite close to 1.

A simple moving average corresponds to a uniform probability distribution and thus its filter width of size n has standard deviation$\sqrt{(n^2-1)/12}$. Thus the application of successive m moving averages with sizes $n_1, \ldots, n_m$ yield a standard deviation of

$$\sigma = \sqrt{\frac{n_1^2 + \cdots + n_m^2 - m}{12}}.$$

(Note that standard deviations do not sum up, but variances do.)

A gaussian kernel requires $6\sigma - 1$ values, e.g. for a σ of 3 it needs a kernel of length 17. A running mean filter of 5 points will have a sigma of $\sqrt{2}$. Running it three times will give a σ of 2.42. It remains to be seen where the advantage is over using a gaussian rather than a poor approximation.

When applied in two dimensions, this formula produces a Gaussian surface that has a maximum at the origin, whose contours are concentric circles with the origin as centre. A two dimensional convolution matrix is precomputed from the formula and convolved with two dimensional data. Each element in the resultant matrix new value is set to a weighted average of that elements neighborhood. The focal element receives the heaviest weight (having the highest Gaussian value) and neighboring elements receive smaller weights as their distance to the focal element increases. In Image processing, each element in the matrix represents a pixel attribute such as brightness or color intensity, and the overall effect is called Gaussian blur.

The Gaussian filter is non-causal which means the filter window is symmetric about the origin in the time-domain. This makes the Gaussian filter physically unrealizable. This is usually of no consequence for applications where the filter bandwidth is much larger than the signal. In real-time systems, a delay is incurred because incoming samples need to fill the filter window before the filter can be applied to the signal. While no amount of delay can make a theoretical Gaussian filter causal (because the Gaussian function is non-zero everywhere), the Gaussian function converges to zero so rapidly that a causal approximation can achieve any required tolerance with a modest delay, even to the accuracy of floating point representation.

B. Histogram construction

A histogram is a graphical representation of the distribution of numerical data. It is an estimate of the probability distribution of a continuous variable (quantitative variable) and was first introduced by Karl Pearson. To construct a histogram, the first step is to "bin" the range of values that is, divide the entire range of values into a series of intervals and then count how many values fall into each interval. The bins are usually specified as consecutive, non-overlapping intervals of a variable. The bins (intervals) must be adjacent, and are usually equal size.

If the bins are of equal size, a rectangle is erected over the bin with height proportional to the frequency, the number of cases in each bin. In general, however, bins need not be of equal width; in that case, the erected rectangle has are a proportional to the frequency of cases in the bin, the vertical axis is not frequency but density: the number of cases per unit of the variable on the horizontal axis. A histogram may also be normalized displaying relative frequencies. It then shows the proportion of cases that fall into each of several categories, with the sum of the heights equalling. Examples of variable bin width are displayed on Census bureau data below.

As the adjacent bins leave no gaps, the rectangles of a histogram touch each other to indicate that the original variable is continuous.

Histograms give a rough sense of the density of the underlying distribution of the data, and often for density estimation: estimating the probability density function of the underlying variable. The total area of a histogram used for probability density is always normalized to 1. If the length of the intervals on the x-axis is all 1, then a histogram is identical to a relative frequency plot.

A histogram can be thought of as a simplistic kernel density estimation, which uses a kernel to smooth frequencies over the bins. This yields a smoother probability density function, which will in general more accurately reflect distribution of the underlying variable. The density estimate could be plotted as an alternative to the histogram, and is usually drawn as a curve rather than a set of boxes.

Another alternative is the average shifted histogram, which is fast to compute and gives a smooth curve estimate of the density without using kernels.

The histogram is one of the seven basic tools of quality control.

Histograms are often confused with bar charts. A histogram is used for continuous data, where the bins represent ranges of data, and the areas of the rectangles are meaningful, while a bar chart is a plot of categorical variables and the discontinuity should be indicated by having gaps between the rectangles, from which only the length is meaningful. Often this is neglected, which may lead to a bar chart being confused for a histogram.

In a more general mathematical sense, a histogram is a function m_i that counts the number of observations that fall into each of the disjoint categories (known as bins), whereas the graph of a histogram is merely one way to represent a histogram. Thus, if we let n be the total number of observations and k be the total number of bins, the histogram m_i meets the following conditions:

$$n = \sum_{i=1}^{k} m_i. \tag{1}$$

C. Cumulative histogram

A cumulative histogram is a mapping that counts the cumulative number of observations in all of the bins up to the specified bin. That is, the cumulative histogram M_i of a histogram m_j is defined as:

$$M_i = \sum_{j=1}^{i} m_j. \tag{1}$$

D. Number of bins and width

There is no "best" number of bins, and different bin sizes can reveal different features of the data. Grouping data is at least as old as Graunt's work in the 17th century, but no systematic guidelines were given until Sturges's work in 1926.

Using wider bins where the density is low reduces noise due to sampling randomness; using narrower bins where the density is high (so the signal drowns the noise) gives greater precision to the density estimation. Thus varying the bin-width within a histogram can be beneficial. Nonetheless, equal-width bins are widely used.

Some theoreticians have attempted to determine an optimal number of bins, but these methods generally make strong assumptions about the shape of the distribution. Depending on the actual data distribution and the goals of the analysis, different bin widths may be appropriate, so experimentation is usually needed to determine an appropriate width. There are, however, various useful guidelines and rules of thumb.

In statistics, kernel density estimation (KDE) is a non-parametric way to estimate the probability density function of a random variable. Kernel density estimation is a fundamental data smoothing problem where inferences about the population are made, based on a finite data sample. In some fields such as signal processing and econometrics it is also termed the Parzen–Rosenblatt window method, after Emanuel Parzen and Murray Rosenblatt, who are usually credited with independently creating it in its current form

This histogram differs from the first only in the vertical scale. The area of each block is the fraction of the total that each category represents, and the total area of all the bars is equal to 1 (the fraction meaning "all"). The curve displayed is a simple density estimate. This version shows proportions, and is also known as a unit area histogram.

In other words, a histogram represents a frequency distribution by means of rectangles whose widths represent class intervals and whose areas are proportional to the corresponding frequencies: the height of each is the average frequency density for the interval. The intervals are placed together in order to show that the data represented by the histogram, while exclusive, is also contiguous. (E.g., in a histogram it is possible to have two connecting intervals of 10.5–20.5 and 20.5–33.5, but not two connecting intervals of 10.5–20.5 and 22.5–32.5. Empty intervals are represented as empty and not skipped.).

E. Discrete Cosine Transform (DCT)

A discrete cosine transform (DCT) expresses a finite sequence of data points in terms of a sum of cosine functions oscillating at different frequencies. DCTs are important to numerous applications in science and engineering, from lossy_compression of audio (e.g. MP3)and images (e.g. JPEG) (where small high-frequency components can be discarded), to spectral methods for the numerical solution of partial differential equations. The use of cosine rather than sine functions is critical for compression, since it turns out (as described below) that fewer cosine functions are needed to approximate a typical signal, whereas for differential equations the cosines express a particular choice of boundary_conditions.

In particular, a DCT is a Fourier-related similar to the discrete Fourier transform (DFT), but using only real numbers. DCTs are equivalent to DFTs of roughly twice the length, operating on real data with even symmetry (since the Fourier transform of a real and even function is real and even), where in some variants the input and/or output data are shifted by half a sample. There are eight standard DCT variants, of which four are common.

The most common variant of discrete cosine transform is the type-II DCT, which is often called simply "the DCT", its inverse, the type-III DCT, is correspondingly often called simply "the inverse DCT" or "the IDCT". Two related transforms are the discrete sine_transforms (DST), which is equivalent to a DFT of real and odd functions, and the modified discrete cosine_transforms (MDCT), which is based on a DCT of overlapping data.

The DCT, and in particular the DCT-II, is often used in signal and image processing, especially for lossy compression, because it has a strong "energy compaction" property: in typical applications, most of the signal information tends to be concentrated in a few low-frequency components of the DCT. For strongly correlated Markov processes, the DCT can approach the compaction efficiency of the Karhunen Loève_transform (which is optimal in the decorrelation sense). As explained below, this stems from the boundary conditions implicit in the cosine functions.

Like any Fourier-related transform, discrete cosine transforms (DCTs) express a function or a signal in terms of a sum of sinusoids with different frequencies and amplitudes. Like the discrete_Fourier transform (DFT), a DCT operates on a function at a finite number of discrete data points. The obvious distinction between a DCT and a DFT is that the former uses only cosine functions, while the latter uses both cosines and sines (in the form of complex exponentials). However, this visible difference is merely a consequence of a deeper distinction: a DCT implies different boundary conditions than the DFT or other related transforms.

Formally, the discrete cosine transform is a linear, invertible function $f : \mathbb{R}^N \to \mathbb{R}^N$ (where $\mathbb{R}$ denotes the set of real numbers), or equivalently an invertible N ×N square matrix. There are several variants of the DCT with slightly modified definitions. The N real numbers x_0, x_{N-1} are transformed into the N real numbers X_0... X_{N-1} according to one of the formulas:

A. DCT-I

$$X_k = \frac{1}{2}(x_0 + (-1)^k x_{N-1}) + \sum_{n=1}^{N-2} x_n \cos\left[\frac{\pi}{N-1} nk\right] \qquad k = 0, \dots, N-1.$$

Some authors further multiply the x_0 and x_{N-1} terms by √2, and correspondingly multiply the X_0 and X_{N-1} terms by 1/√2. This makes the DCT-I matrix orthogonal, if one further multiplies by an overall scale factor of $\sqrt{2/(N-1)}$, but breaks the direct correspondence with a real-even DFT.

The DCT-I is exactly equivalent (up to an overall scale factor of 2), to a DFT of $2N-2$ real numbers with even symmetry. For example, a DCT-I of N=5 real numbers abcde is exactly equivalent to a DFT of eight real numbers abcdedcb (even symmetry), divided by two. (In contrast, DCT types II-IV involve a half-sample shift in the equivalent DFT.)

Note, however, that the DCT-I is not defined for N less than 2. (All other DCT types are defined for any positive N.)

Thus, the DCT-I corresponds to the boundary conditions: x_n is even around n=0 and even around n=N-1; similarly for X_k.

B. DCT-II

$$X_k = \sum_{n=0}^{N-1} x_n \cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right] \qquad k=0,\ldots,N-1.$$

The DCT-II is probably the most commonly used form, and is often simply referred to as "the DCT".

This transform is exactly equivalent (up to an overall scale factor of 2) to a DFT of $4N$ real inputs of even symmetry where the even-indexed elements are zero. That is, it is half of the DFT of the $4N$ inputs y_n, where $y_{2n}=0$, $y_{2n+1}=x_n$ for $0 \leq n < N$, $y_{2N}=0$, and $y_{4N-n}=y_n$ for $0 < n < 2N$.

Some authors further multiply the X_0 term by $1/\sqrt{2}$ and multiply the resulting matrix by an overall scale factor of $\sqrt{2/N}$ (see below for the corresponding change in DCT-III). This makes the DCT-II matrix orthogonal, but breaks the direct correspondence with a real-even DFT of half-shifted input. This is the normalization used by Matlab, for example. In many applications, such as JPEG, the scaling is arbitrary because scale factors can be combined with a subsequent computational step (e.g. the quantization step in JPEG), and a scaling that can be chosen that allows the DCT to be computed with fewer multiplications.

The DCT-II implies the boundary conditions: x_n is even around n=-1/2 and even around n=N-1/2; X_k is even around k=0 and odd around k=N.

The Fourier-related transforms that operate on a function over a finite domain, such as the DFT or DCT or a Fourier series, can be thought of as implicitly defining an extension of that function outside the domain. That is, once you write a function $f(x)$ as a sum of sinusoids, you can evaluate that sum at any x, even for x where the original $f(x)$ was not specified. The DFT, like the Fourier series, implies a periodic extension of the original function. A DCT, like a cosine transform, implies an even extension of the original function.

In particular, it is well known that any discontinuities in a function reduce the rate of convergence of the Fourier series, so that more sinusoids are needed to represent the function with a given accuracy. The same principle governs the usefulness of the DFT and other transforms for signal compression: the smoother a function is, the fewer terms in its DFT or DCT are required to represent it accurately, and the more it can be compressed. (Here, we think of the DFT or DCT as approximations for the Fourier series or cosine series of a function, respectively, in order to talk about its "smoothness".) However, the implicit periodicity of the DFT means that discontinuities usually occur at the boundaries: any random segment of a signal is unlikely to have the same value at both the left and right boundaries. (A similar problem arises for the DST, in which the odd left boundary condition implies a discontinuity for any function that does not happen to be zero at that boundary.) In contrast, a DCT where both boundaries are even always yields a continuous extension at the boundaries (although the slope is generally discontinuous). This is why DCTs, and in particular DCTs of types I, II, V, and VI (the types that have two even boundaries) generally perform better for signal compression than DFTs and DSTs. In practice, a type-II DCT is usually preferred for such applications, in part for reasons of computational convenience.

Disadvantages

- It is complex
- It has poor energy compaction

2. **Proposed Methods**

In existing method we have used discrete wavelet transformation method which is used for the image only with high accuracy to overcome this drawback we are here introducing a method called lifting wavelet transform which is used to extract or transfer the images with both high and low accuracy.

A. Lifting Wavelet Transform:

The lifting scheme is a technique for both designing wavelets and performing the discrete wavelet transform. Actually it is worthwhile to merge these steps and design the wavelet filters while performing the wavelet transform. This is then called the second generation wavelet transform. The technique was introduced by Wim Sweldens.

The discrete wavelet transform applies several filters separately to the same signal. In contrast to that, for the lifting scheme the signal is divided like a zipper. Then a series of convolution-accumulate operations across the divided signals is applied.

The basic idea of lifting is the following: If a pair of filters (h, g) is *complementary*, that is it allows for *perfect reconstruction*, then for every filter s the pair (h', g) with $h'(z) = h(z) + s(z^2) \cdot g(z)$ allows for perfect reconstruction, too. Of course, this is also true for every pair (h, g') of the form $g'(z) = g(z) + t(z^2) \cdot h(z)$. The converse is also true: If the filter banks (h, g) and (h', g) allow for perfect reconstruction, then there is a unique filter s with $h'(z) = h(z) + s(z^2) \cdot g(z)$.

Each such transform of the filter bank (or the respective operation in a wavelet transform) is called a lifting step. A sequence of lifting steps consists of alternating lifts, that is, once the lowpass is fixed and the highpass is changed and in the next step the highpass is fixed and the lowpass is changed. Successive steps of the same direction can be merged.

- Perfect reconstruction
- Every transform by the lifting scheme can be inverted.
- Every perfect reconstruction filter bank can be decomposed into lifting steps by the Euclidean algorithm.
- That is, "lifting decomposable filter bank" and "perfect reconstruction filter bank" denotes the same.
- Every two perfect reconstructable filter banks can be transformed into each other by a sequence of lifting steps. (If P and Q are polyphase matrices with the same determinant, the lifting sequence from P to Q, is the same as the one from the lazy polyphase matrix I to $P^{-1} \cdot Q$.)
- Speedup by a factor of two: This is only possible because lifting is restricted to perfect reconstruction filterbanks. That is, lifting somehow squeezes out redundancies caused by perfect reconstructability.
- In place: The transformation can be performed immediately in the memory of the input data with only constant memory overhead.
- Non-linearities: The convolution operations can be replaced by any other operation. For perfect reconstruction only the invertibility of the addition operation is relevant. This way rounding errors in convolution can be tolerated and bit-exact reconstruction is possible. However the numeric stability may be reduced by the non-linearities. This must be respected if the transformed signal is processed like in lossy compression.

Although every reconstruct able filter bank can be expressed in terms of lifting steps, a general description of the lifting steps is not obvious from a description of a wavelet family. However, for instance for simple cases of the Cohen-Daubechies-Feauveau wavelet, there is an explicit formula for their lifting steps.

B. Embedding method

A digital watermarking method is referred to as spread-spectrum if the marked signal is obtained by an additive modification. Spread-spectrum watermarks are known to be modestly robust, but also to have a low information capacity due to host interference.

A digital watermarking method is said to be of quantization type if the marked signal is obtained by quantization. Quantization watermarks suffer from low robustness, but have a high information capacity due to rejection of host interference.

A digital watermarking method is referred to as amplitude modulation if the marked signal is embedded by additive modification which is similar to spread spectrum method, but is particularly embedded in the spatial domain.

A digital watermark is called robust with respect to transformations if the embedded information may be detected reliably from the marked signal, even if degraded by any number of transformations. Typical image degradations are JPEG compression, rotation, cropping, additive noise, and quantization. For video content, temporal modifications and MPEG compression often are added to this list. A digital watermark is called imperceptible if the watermarked content is perceptually equivalent to the original, unwatermarked content. In general, it is easy to create either robust watermarks or imperceptible watermarks, but the creation of both robust and imperceptible watermarks has proven to be quite challenging. Robust imperceptible watermarks have been proposed as a tool for the protection of digital content, for example as an embedded no-copy-allowed flag in professional video content.

Digital watermarking techniques may be classified in several ways.

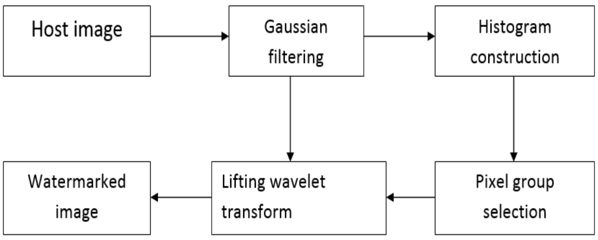

Figure 1. Watermark Embedding and Encoding Process

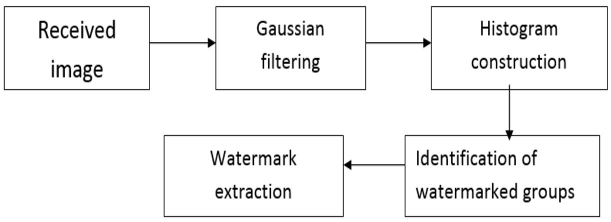

Figure 2. Watermark Decoding

III. RESULTS AND DISCUSSION

A. Robustness

A digital watermark is called "fragile" if it fails to be detectable after the slightest modification. Fragile watermarks are commonly used for tamper detection (integrity proof). Modifications to an original work that clearly are noticeable commonly are not referred to as watermarks, but as generalized barcodes.

A digital watermark is called semi-fragile if it resists benign transformations, but fails detection after malignant transformations. Semi-fragile watermarks commonly are used to detect malignant transformations.

A digital watermark is called robust if it resists a designated class of transformations. Robust watermarks may be used in copy protection applications to carry copy and no access control information.

B. Perceptibility

A digital watermark is called imperceptible if the original cover signal and the marked signal are perceptually indistinguishable.

A digital watermark is called perceptible if its presence in the marked signal is noticeable (e.g. Digital On-screen Graphics like a Network Logo, Content Bug, Codes, Opaque images). On videos and images, some are made transparent/translucent for convince for people due to the fact that they block portion of the view.

This should not be confused with perceptual, that is, watermarking which uses the limitations of human perception to be imperceptible.

C. Capacity

The length of the embedded message determines two different main classes of digital watermarking schemes:

- The message is conceptually zero-bit long and the system is designed in order to detect the presence or the absence of the watermark in the marked object. This kind of watermarking scheme is usually referred to as zero-bit or presence watermarking schemes. Sometimes, this type of watermarking scheme is called 1-bit watermark, because a 1 denotes the presence (and a 0 the absence) of a watermark.
- The message is an n-bit-long stream $(m = m_1 \ldots m_n,\ n \in \mathbb{N}$ with $n = |m|)$ or $M = \{0,1\}^n$ and is modulated in the watermark. These kinds of schemes usually are referred to as multiple-bit watermarking or non-zero-bit watermarking schemes.

Reversible data hiding is a technique which enables images to be authenticated and then restored to their original form by removing the digital watermark and replacing the image data that had been overwritten. This would make the images acceptable for legal purposes. The U.S. Army also is interested in this technique for authentication of reconnaissance images.

Table 1. PSNR

Methods	PSNR(db)
Histogram shifting technique	72.41

Haar wavelet transform	78.62
Sorting technique	74.81
Optimal histogram pair shifting	78.00
Dynamic prediction error histogram shifting	79.06
Duality approach	39.10
DWT-DFT composite watermarking	41.10
DWT method	41.1
DCT	26.44
Non-Linear Regression	8.45

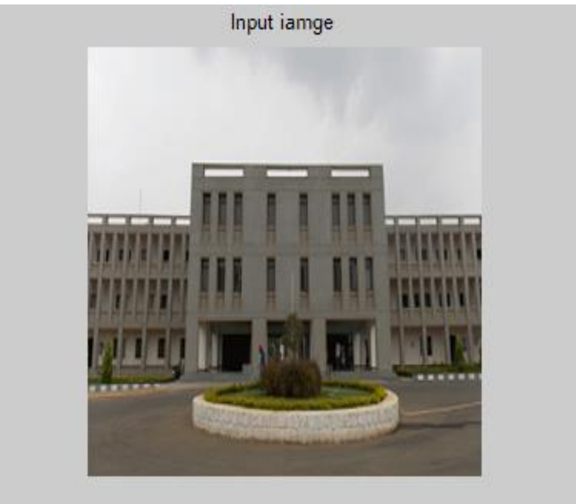

Figure 1. Input Image

Figure 2. Gray Image

Figure 3. Watermarked Image

Figure 4. Watermarked with LWT

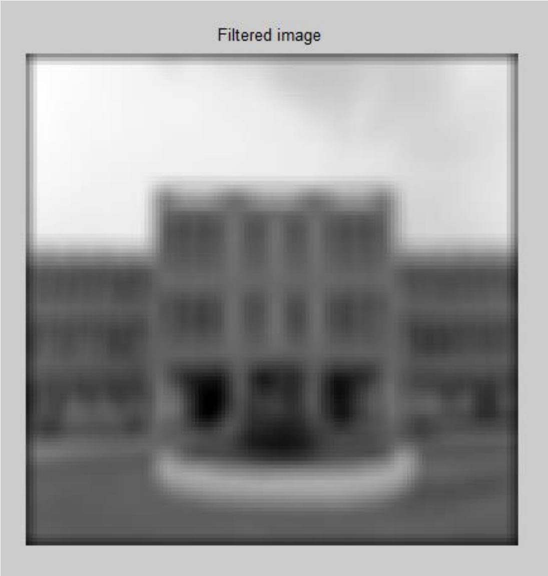

Figure 5. Filtered Image

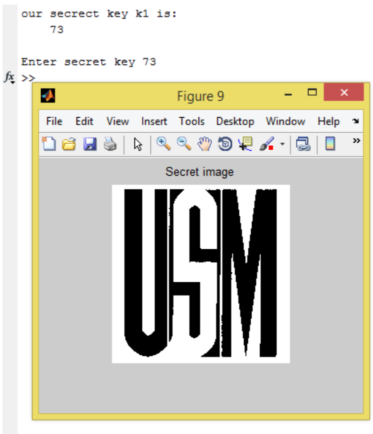

Figure 6. Secret Image

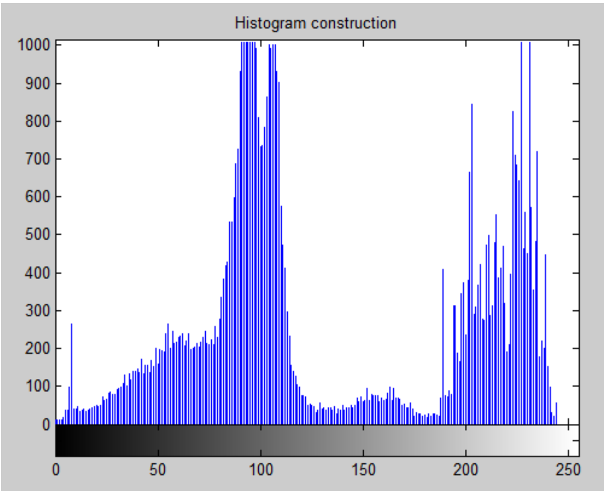

Figure 7. Histogram Construction for Host Image

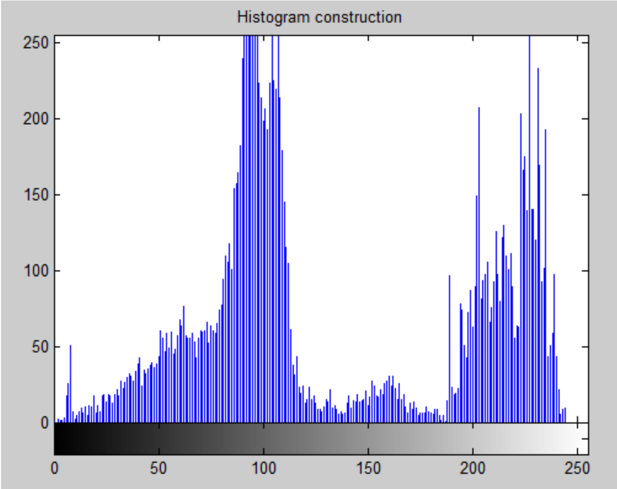

Figure 8. Histogram Construction for Secret Image

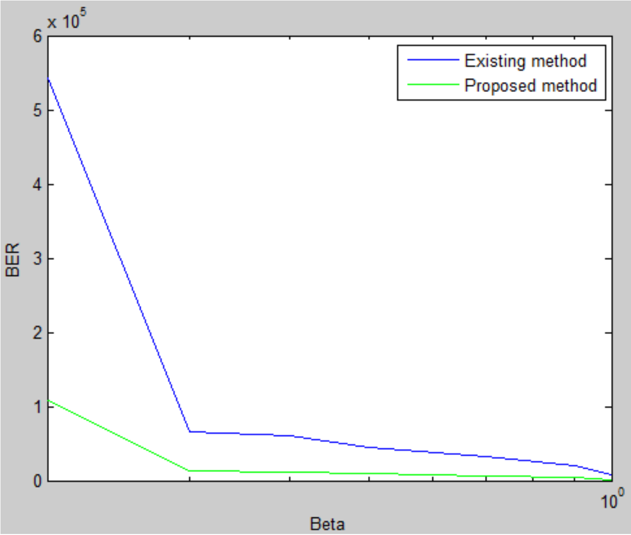

Figure 9. BER

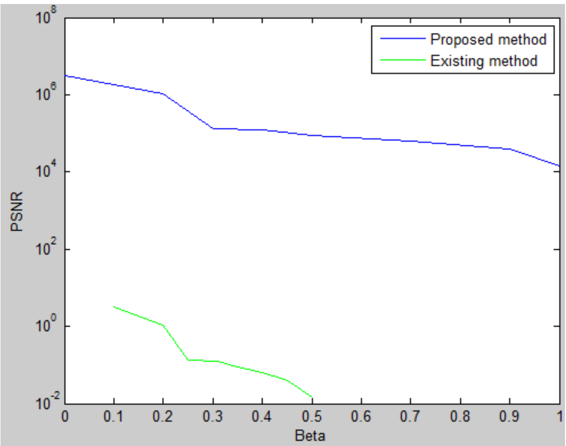

Figure 10. PSNR-Beta

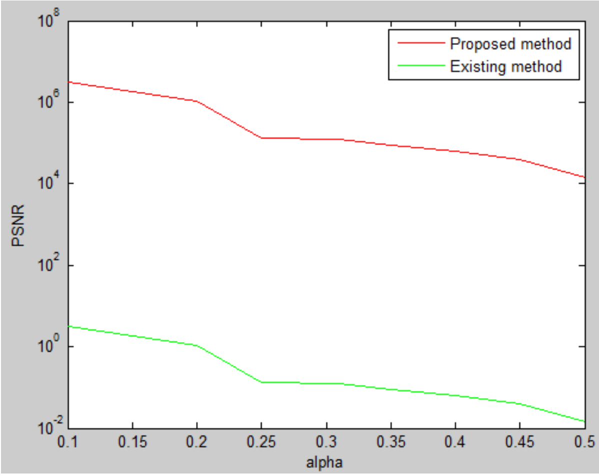

Figure 11. PSNR-Alpha

IV. CONCLUSION

The attacks are removed using, a Gaussian low-pass filter which is employed to pre-process the host image such that watermarks will only be embedded into the High frequency component of the host image. To tackle the attacks like cropping attacks, a histogram, shape, related index is utilized to form and select the most suitable pixel groups for watermark embedding. In addition, a safe band is introduced between the selected pixel groups and the non-selected pixel groups to improve robustness. In this paper, LWT is proposed to improve the robustness of the host image. It can be further verified to improve the efficiency to extract the secret image from the host image. Due to the usage of secret key, the proposed watermarking method is also secure.

V. REFERENCES

[1] Y. Xiang, I. Natgunanathan, D. Peng, W. Zhou, and S. Yu, “A dualchannel time-spread echo method for audio watermarking,” IEEE Trans. Inf. Forensics Security, vol. 7, no. 2, pp. 383–392, Apr. 2012.

[2] Y. Xiang, D. Peng, I. Natgunanathan, and W. Zhou, “Effective pseudonoise sequence and decoding function for imperceptibility and robustness enhancement in time-spread echo-based audio watermarking,” IEEE Trans. Multimedia, vol. 13, no. 1, pp. 2–13, Feb. 2011.

[3] N. K. Kalantari, S. M. Ahadi, and M. Vafadust, “A robust image watermarking in the ridgelet domain using universally optimum decoder,” IEEE Trans. Circuits Syst. Video Technol., vol. 20, no. 3, pp. 396–406, Mar. 2010.

[4] E. Nezhadarya, Z. J. Wang, and R. K. Ward, “Robust image watermarking based on multiscale gradient direction quantization,” IEEE Trans. Inf. Forensics Security, vol. 6, no. 4, pp. 1200–1213, Dec. 2011.

[5] L. Wang, H. Ling, F. Zou, and Z. Lu, “Real-time compressed domain video watermarking resistance to geometric distortions,” IEEE MultiMedia, vol. 19, no. 1, pp. 70–79, Jan. 2012.

[6] M. Lee, K.-S. Kim, and H.-K. Lee, “Digital cinema watermarking for estimating the position of the pirate,” IEEE Trans. Multimedia, vol. 12, no. 7, pp. 605–621, Nov. 2010.

[7] N. F. Johnson, Z. Duric, and S. Jajodia, “Recovery of watermarks from distorted images,” in Proc. 3rd Int. Workshop Inf. Hiding, 1999, pp. 318–332.

[8] Wenpeng Ding, Feng Wu, Senior Member, IEEE, Xiaolin Wu, Senior Member, IEEE, Shipeng Li, Member, IEEE, and Houqiang Li, “Adaptive Directional Lifting-Based Wavelet Transform for Image Coding” IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 16, NO. 2, FEBRUARY 2007.

[9] Zdenek Prusa and Pavel Rajmic “Real-Time lifting wavelet transform algorithm”, elekrt@revue VOL. 2, NO. 3, SEPTEMBER 2011.

[10] Chaveli Ramesh, T. Ranjith,“Fusion Performance Measures and a Lifting Wavelet Transform based algorithm for Image Fusion”, ISIF- 2002.

[11] J. F. Lichtenauer, I. Setyawan, T. Kalker, and R. L. Lagendijk, “Exhaustive geometrical search and the false positive watermark detection probability,” Proc. SPIE, Secur. Watermarking Multimedia Contents V, vol. 5020, pp. 203–214, Jun. 2003.

[12] M. Barni, “Effectiveness of exhaustive search and template matching against watermark desynchronization,” IEEE Signal Process. Lett., vol. 12, no. 2, pp. 158–161, Feb. 2005.

[13] J. J. K. Ó. Ruanaidh and T. Pun, “Rotation, scale and translation invariant spread spectrum digital image watermarking,” Signal Process., vol. 66, no. 3, pp. 303–317, May 1998.

[14] C.-Y. Lin, M. Wu, J. A. Bloom, I. J. Cox, M. L. Miller, and Y. M. Lui, “Rotation, scale, and translation resilient watermarking for images,” IEEE Trans. Image Process., vol. 10, no. 5, pp. 767–782, May 2001.