An Attack-resistant Watermark Resynchronization Scheme using LDFT and BSP

Meeramol T K, Haripriya Nair
Dept. of CSE Ilahia College of Engineering & Technology Kerala, India
Assistant Professor Dept. of CSE Ilahia College of Engineering & Technology Kerala, India

Abstract: *Image watermarking is a method that embeds a watermark in the digital image by making small changes in the host data. In watermarking applications, the robustness of the watermark to the common signal processing and geometric desynchronization attacks(DAs) is essential to the system. Most image watermarking resynchronization schemes can survive global DAs (e.g., rotation, scaling, translation, and other affine transforms), but few are resilient to cropping and local DAs. In this paper, We present a watermarking resynchronization scheme against local transform attack. It uses a local invariant transform called Local Daisy Feature Transform(LDFT) and Binary Space Partitioning(BSP) Tree to partition the LDFT space. Watermark is embedded in the leaf nodes of the BSP tree. Here, attacks to the watermarked image can also be detected using SHA-1 algorithm.*

Keywords: *desynchronization attack ; resynchronization ; transform.*

I. Introduction

With the wide range of rapidly growing technologies of the internet and digital media, visual data such as images can easily be copied, altered and distributed over the internet without any loss in quality. These unauthorized activities are potentially capable of incurring considerable financial loss to the media producers and content providers. Therefore the protection of the ownership of multimedia data has become a very challenging issue. As an effective and efficient solution, image watermarking is introduced. Image watermarking superimposes a copyright message into a host image before dissemination and then unauthorized reproduction can be recognized by extracting the copyright information.

Along with the rapid growth of watermarking schemes, various attacking attempts have also been developed to destroy watermarks. Among these attacks, geometric attacks are very difficult to handle. This is mainly due to the fact that slight geometric manipulation to the marked image, such as scaling or rotation, could significantly reduce the possibility of a successful watermark retrieval, provided that the watermarking extractor has no knowledge of the distortion parameters. In another word, geometric distortion can easily introduce synchronization errors into the watermark extraction process. In recent years, a number of approaches have been proposed to counteract the desynchronization attacks.

There are several watermarking techniques ,robust against desynchronization attacks have been proposed. [2]describes how a combination of spread spectrum encoding of the embedded message and transform-based invariants can be used for digital image watermarking .In particular, it is described how a Fourier-Mellin-based approach can be used to construct watermarks which are designed to be unaffected by any combination of rotation and scale transformations. [3]combines image feature extraction and image normalization is proposed. A feature extraction method called Mexican Hat wavelet scale interaction is used. The extracted feature points can survive a variety of attacks and be used as reference points for both watermark embedding and detection.

In [4] two watermarking approaches that are robust to geometric distortions are presented. The first approach is based on image normalization, in which both watermark embedding and extraction are carried out with respect to an image normalized to meet a set of predefined moment criteria. The second approach is based on a watermark resynchronization scheme, which uses deformable mesh, aimed to alleviate the effects of random bending attacks. [5] proposes three different invariant-region detection methods based on the scale-space representation of an image , which are considered for watermarking. At each invariant region, the watermark is embedded after geometric normalization according to the shape of the region.

[6] presents a genetic algorithm based methodology by adjusting gray values of a cover-image while creating the desired statistic features to generate the stego-images that can break the inspection of steganalytic systems. [7] propose an original blind watermarking algorithm robust to local geometrical distortions such as the deformations induced by Stirmark. This method consists in adding a predefined additional information to the useful message bits at the insertion step. These additional bits are labeled as resynchronization bits or reference bits and they are modulated in the same way as the information bits.

[8] introduce a local image descriptor, DAISY, which is very efficient to compute densely. I used a novel local invariant feature transform, named LDFT, which is inspired by the DAISY descriptor. In [9], the concept of the Random Jitter Attack is introduced. This attack is characterized by the displacement of each pixel position by a random amount given by an arbitrary distribution, followed by interpolation over the modified sampling grid.[10] evaluates different feature extraction techniques and it shows that the scale-invariant keypoint extractor is appropriate for robust watermarking.

In [11], a novel arrangement for quantizer levels in the Quantization Index Modulation (QIM) method is proposed. The compression function of μ-Law standard is used for quantization. In this regard, the host signal is first transformed into the logarithmic domain using the μ-Law compression function. Then, the transformed data is quantized uniformly and the result is transformed back to the original domain using the inverse function.

In this paper, I present a watermarking resynchronization scheme, which is resilient to various attacks, including local and global DAs and noninvertible cropping. A new feature transform named local daisy feature transform (LDFT) is used , and each pixel can be mapped into the LDFT space. The geometrically invariant LDFT space is then partitioned with the binary space partitioning (BSP) tree. The watermarking sequence is embedded bit by bit into each leaf node of the BSP tree by using the logarithmic quantization index modulation (LQIM) [20] watermarking embedding method. At the receiver side , the tampering can also be detected using Secure Hash Algorithm-1(SHA-1) .

II. Related Works

System security is based on proprietary knowledge of the keys which are required to embed, extract or remove an image watermark. It is important that the watermarking process incorporate some non-invertible step which may depend on a private key or a hash function of the original image. Only in this way can true ownership of the copyright material be resolved. Fourier-Mellin-based approach can be used to construct watermarks which are designed to be unaffected by any combination of rotation and scale transformations. In addition, a CDMA spread spectrum encoding allows one to embed watermark messages of arbitrary length and which need only a secret key for decoding[2]. A combination of spread spectrum encoding of the embedded message and transform-based invariants can be used for digital image watermarking. Spread spectrum is an example of a symmetric key cryptosystem. CDMA is a method for encoding binary messages which can later be recovered given knowledge of the key used.

A feature based robust watermarking scheme[3] use a combination of image feature extraction and image normalization. A feature extraction method called Mexican Hat wavelet scale interaction is used here. The extracted feature points are used for watermark embedding and detection. Discs are placed at the feature points. Several copies of a 16-bit watermark sequence are embedded in the normalized discs in the original image to improve the robustness of watermarks. These reference points can also act as marks for (location) synchronization between watermark embedding and detection. Since Mexican Hat wavelet scale interaction is formed by two scales, it allows different degrees of robustness. Since local variations such as cropping or warping generally affect only a few feature points in an image, the unaffected feature points can still be used as references during the detection process. These feature points are the centers of the disks that are to be used for watermark embedding. Several geometric central moments are computed to transform the input image to its normalized form. The normalized image (object) of a rotated image (object) is the same as the normalized image of the original image. The performance of this scheme is limited by the robustness of the feature points.

Two watermarking approaches together can be used ,that are robust to geometric distortions[4]. That include image normalization and image watermarking resynchronization scheme. It is to alleviate the effects of random bending attacks. In this scheme, a deformable mesh is used to correct the distortion caused by the attack. In both schemes, they employ a direct-sequence code division multiple access approach to embed a multibit watermark in the discrete cosine transform domain of the image. The first is a multibit public watermarking scheme based on image normalization, aimed to be robust to general affine geometric attacks. The second watermarking approach is based on a watermark resynchronization scheme, aimed to be robust to random geometric distortions and to be used in the context of private watermarking where the original image is known. The original image and the potentially attacked watermarked image are used to estimate a mesh model of the unknown geometric distortion. This scheme is robust against Random Bending Attacks.

A content-based image watermarking method[5] based on invariant regions of an image is robust against various image processing steps, including geometric transformations, cropping, filtering, and JPEG compression. At each invariant region, the watermark is embedded after geometric normalization according to the shape of the region. Three invariant-region detection methods based on the scale-space representation of an image were considered for watermarking. The most obvious way to achieve resilience against geometric distortions is to use an invariant transform. A feature point provides only the position information. To cope with affine or projective transformations, we need additional information about the geometric transformations, which can be acquired from the neighborhood of the feature point. Three different methods are considered for that purpose, such as characteristic scale, shape adaptation of the Gaussian kernel, and feature-point sets. characteristic scale of a feature point is used for scale-invariant pattern matching. The characteristic scale is the scale at which the normalized scale-space representation of an image attains a maximum value. Shape adaptation procedure that adapts the Gaussian smoothing kernel to the local image structures around the feature point. The shape adaptation process is iteratively performed until the shape of the kernel converges. Feature-point sets obtained by grouping feature points are used for invariant pattern matching. The watermark is shaped adaptively based on the invariant regions of an image. The Harris points appropriate for watermarking are extracted from the scale-space representation of an image. Invariant regions are generated at each scale based on characteristic scale, shape adaptation of the Gaussian kernel, and feature-point sets. The scale refers to the standard deviation of the associated Gaussian kernel. The complex geometric shapes require more computation in both the invariant-region detection and the local search for watermark detection. However, the watermarks with complex geometric shapes perform better for the nonisotropic scaling and the projective transformations.

A steganalytic system is able to detect stego-images. A genetic algorithm based methodology[6] provide a new concept for developing robust steganographic system by using statistic features instead of the traditional strategy by avoiding the change of statistic features. The genetic algorithm based methodology adjusts gray values of a cover-image while creating the desired statistic features to generate the stego-images that can break the inspection of steganalytic systems. It increase the capacity of the embedded message and enhance the peak signal-to-noise ratio of stego-images. In general, the GA starts with some randomly selected genes as the first generation, called population. Each individual in the population corresponding to a solution in the problem domain is called chromosome. An objective, called fitness function, is used to evaluate the quality of each chromosome. The chromosomes of high quality will survive and form a new population of the next generation. By using the three operators: reproduction, crossover, and mutation, we recombine a new generation to find the best solution. The process is repeated until a predefined condition is satisfied or a constant number of iterations are reached. This is applied for embedding messages into the frequency domain of a cover image to obtain the stego-image. GA-based algorithm can be used to enhance the quality of stego-images.

Still image watermarking[7] is the original blind watermarking algorithm robust to local geometrical distortions such as the deformations induced by Stirmark. This method uses the insertion of a predefined additional information to the useful message bits at the insertion step. These additional bits are labeled as resynchronization bits or reference bits and they are modulated in the same way as the information bits. This paper present an original approach to compensate for local and global geometrical distortions, in particular the random deformations generated by the Stirmark attack. This method operates directly in the spatial domain, and does not require for the extraction step any a priori knowledge neither on the original image, nor about the hidden message. During the extraction step, the resynchronization bits are used as reference points to estimate and compensate for small local or global geometrical deformations. The method is able to compensate for all the geometrical deformations which can be locally approximated by block matching.

DAISY[8] is a local image descriptor, which is very efficient to compute densely. It is inspired from earlier ones such as SIFT and GLOH but can be computed much faster for our purposes. Unlike SURF which can also be computed efficiently at every pixel, it does not introduce artifacts that degrade the matching performance when used densely. This descriptor can be used for dense matching and view-based synthesis using stereo-pairs having various image transforms or for pairs with too large a baseline for standard correlation-based techniques to work. At each pixel location, DAISY consists of a vector made of values from the convolved orientation maps located on concentric circles centered on the location, and where the amount of Gaussian smoothing is proportional to the radii of the circles. This gives the descriptor the appearance of a flower, hence its name. Speed increase comes from replacing weighted sums used by the earlier descriptors by sums of convolutions, which can be computed very quickly and from using a circularly symmetrical weighting kernel.

Effects of synchronization errors on the bit error rate performance of spatial domain image watermarking is studied in[9]. For that, Random Jitter Attack is considered. This attack is characterized by the displacement of each pixel position by a random amount given by an arbitrary distribution, followed by interpolation over the modified sampling grid. A channel model is proposed to describe the effects of the jitter attack on the watermarked image. This model suggests that the effects of the jitter attack can be analyzed as the addition of signal dependent noise to the watermarked image. Attempting to invert geometric attacks by estimating the attack parameters and then performing an inverse transformation on the watermarked image may result in residual synchronization errors due to the inherent uncertainty associated with parameter estimation. The effect of these residual errors in watermark capacity was studied for both the spread spectrum and the Scalar Costa's Scheme methods, under the simplifying assumption of a constant offset in the sampling grid applied over all the sampling grid . Even in that simple case, a significant capacity loss was observed in both methods for very small jitter values, which indicate that synchronization errors may severely degrade the bit error rate (BER) during decoding.

Feature extraction techniques for watermarking is evaluated in [10]. Geometric distortion attacks desynchronize the location of the inserted watermark and hence prevent watermark detection. Watermark synchronization, which is a process of finding the location for watermark insertion and detection, is crucial to design robust watermarking. Generally feature extraction techniques are used in 2nd generation image watermarking schemes. The Harris corner detector and the Mexican Hat wavelet scale interaction method are two of such techniques. . Here, the scale-invariant keypoint extractor is evaluated. After feature extraction, the set of triangles is generated by Delaunay tessellation. These triangles are the location for watermark insertion and detection. Redetection ratio of triangles is evaluated against geometric distortion attacks as well as signal processing attacks. Correlation- based detector is used to determine whether or not the watermark is inserted. Because the watermark is inserted multiple times into the image, it is highly likely that this method has high probability to detect the watermark even after attacks.

Logarithmic quantization Index Modulation[11] technique is a technique used for watermark embedding in an efficient way. Due to the advantages of logarithmic quantization, and to avoid the problems of previous methods, the compression function of µ-Law standard is used for quantization. For achieving this , the host signal is first transformed into the logarithmic domain using the µ-Law compression function. Then, the transformed data is quantized uniformly and the result is transformed back to the original domain using the inverse function.

According to QIM, the watermark data is embedded by quantizing the host signal features using a set of quantizers, each of which associated with a different message. QIM is a blind method in which the original signal is not needed to extract the watermark data. Also, the embedding and extraction functions are simple and easy to implement. The main problem of QIM is designing codebooks of the quantizers.

Inspired by µ-Law concept, the host signal was transformed into logarithmic domain using a compression function. Then, data was embedded into the transformed signal using uniform quantization and the quantized data was transformed into the original domain using inverse function. Due to the use of logarithmic function, smaller step sizes are devoted to smaller amplitudes and larger step sizes are associated with larger amplitudes. Therefore, in comparison with UQIM, this method poses perceptual advantages that lead to stronger watermark insertion.

III. Proposed Method

The proposed system consist of the calculation of LDFT space, which is locally and globally invariant.After that, a binary space partitioning tree is used convert that into a tree and to the leaf nodes of the tree, watermark bits are embedded. Additionally, a hash calculation is performed to detect whether the image has undergone any tampering. For that, Secret hash Algorithm-1(SHA-1) is used. The Fig 1. shows the architecture of proposed method.

The proposed method starts with the calculation of LDFT , which consists of several computation steps on the image to be watermarked. After an LDFT space is generated, it need to be converted to a binary space partitioning tree by selecting different partition planes at each step. The watermark to be embedded is converted to bit format and each of the bits is embedded into the leaf nodes of the tree using logarithmic quantization index modulation technique. Thus , the watermark for authenticity is embedded. So by using this scheme, watermark can be correctly extracted at the receiver side and he can verify the ownership of the received document.

The watermark embedding process consists of three main steps:

1) constructing the feature space with LDFT;
2) partitioning the LDFT space with the BSP tree; and
3) embedding watermark information by LQIM watermark embedding method.

The watermark extracting process resembles watermark embedding, which comprises three main steps:

1) constructing the feature space of LDFT;
2) partitioning the LDFT space with the BSP tree; and
3) watermark extraction.

For checking the integrity of watermarked image, at embedding side, SHA-1 algorithm is used to calculate the hash of the watermarked image, so that the attacks can be detected at the receiver side. Abbreviations and Acronyms

A. Constructing the feature space with LDFT

The proposed watermarking scheme is can survive global and local desynchronization attacks. That is achieved by using a locally and globally invariant feature transform of the image for watermark embedding. The feature transform used here is Local Daisy Feature transform(LDFT), which is based on DAISY descriptor in[8]. LDFT is more robust, because DAISY is not RST invariant , but LDFT is globally and locally RST invariant. The following are the major steps of computation used to generate the LDFT.

Step 1) Compute the characteristic scale map.

For an input image I, LoG operator is used to obtain the characteristic scale map S, which varies proportionally with image scaling. The map S will be used to control the size of local regions for LDFT in step 5). The same content region is used for LDFT even if the image is zoomed and the LDFT is scaling invariant. The LoG for pixel f(x, y) in image I is defined as

$LoG(x, y, \delta i) = \delta_i^2 \ |Lxx(x, y, \delta i) + Lyy(x, y, \delta i)|$

where $L(x, y,\delta i)=G(x, y, \delta i)*f(x, y)$, $Lxx(x, y, \delta i)=\partial^2 L(x,y, \delta i)/\partial x^2$, and $Lyy(x, y, \delta i) = \partial^2 L(x, y, \delta i)/\partial y^2$. $G(x, y, \delta i)$ is the Gaussian kernel with standard deviation δi and mean zero.Given a set of scales δ, the characteristic scale S(x, y) is the scale at which LoG attains a local maximum as the following equation:

$S(x, y) = \text{argmax}_{\delta i \in \delta} \{LoG(x, y, \delta i)\}$

Step 2) Compute the orientation maps.

The orientation maps are defined as $Go = \max(\partial I/\partial o, 0)$, where I is an image and $o(o \in [1,H])$ is the orientation of its erivative. For an input image, we compute H orientation maps for every quantized direction. Go(u, v) is equal to the image gradient at location (u, v) for direction o if it is bigger than zero; otherwise, it is equal to zero. This can preserve the polarity of the intensity change.

Step 3) Compute the convolved orientation maps.

The convolved orientation maps are defined as $G^{\Sigma}_{o} = G^{\Sigma} * \max(\partial I/\partial o, 0)$, where GΣ is a Gaussian kernel and o is the orientation of the derivative. Each orientation map is then convolved with Gaussian kernels of different standard deviation Σs to obtain convolved orientation maps for differently sized regions.

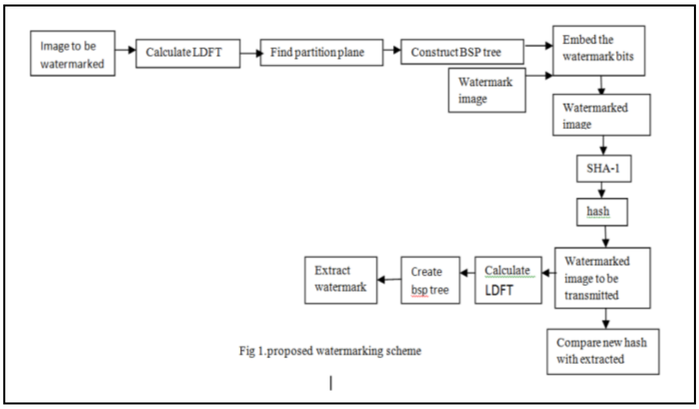


Fig 1.proposed watermarking scheme

Step 4) Normalize the convolved orientation maps of each sample point.

The LDFT consists of a vector made of values from the convolved orientation maps located on concentric circles centered at the location, where the amount of Gaussian smoothing is proportional to the radii of the circles. Let $h_{\Sigma}(x, y)$ be the vector made of the values at location (x, y) in the orientation maps after convolution by a Gaussian kernel of standard deviation Σ.

$\mathbf{h}^{\Sigma}(x, y) =G^{\Sigma}_{1} (x, y), \ldots , G^{\Sigma}_{o} (x, y), \ldots , G^{\Sigma}_{H}(x, y)$

where G^{Σ}_{1} , G^{Σ}_{2} , and G^{Σ}_{H} denote the Σ-convolved orientation maps. Note that H = 36 when 36 orientations are considered. We normalize these vectors to unit norm and denote the normalized vectors by $\tilde{\mathbf{h}}_{\Sigma}(x, y)$.

$\tilde{h}^{\Sigma}(x, y) =[G_{\Sigma}^{1} (x, y), \ldots , \tilde{G}^{\Sigma}_{o} (x, y), \ldots , \tilde{G}^{\Sigma}_{H}(x, y)]$.

Step 5) Compute Euclidean distance.

$\tilde{\mathbf{h}}^{\Sigma}(x, y)$ is a normalized vector, so $\sum_{o=1}^{H} (\tilde{G}^{\Sigma}_{o} (x, y))^2/H=1/H$. We compute the Euclidean distance between the normalized vector $\tilde{\mathbf{h}}_{\Sigma}(x, y)$ and the vector [1/H, . . . , 1/H, . . . , 1/H] and denote the Euclidean distance by $E_{\Sigma}(x, y)$. Note that, if all values of vector $\tilde{\mathbf{h}}_{\Sigma}(x, y)$ are zero, $E_{\Sigma}(x, y)=1/\sqrt{H}$. IfQ represents the number of circular layers, the initial LDFT $\hat{}$ D(x0, y0) for pixel point (x0, y0) is defined as the concatenationof E_{Σ} .

$$\begin{aligned}
&\hat{\mathcal{D}}(x_0, y_0) = [E_{\Sigma_1}(x_0, y_0),\\
&E_{\Sigma_1}(l_1(x_0, y_0, R_1)), \ldots, E_{\Sigma_1}(l_T(x_0, y_0, R_1)),\\
&E_{\Sigma_2}(l_1(x_0, y_0, R_2)), \ldots, E_{\Sigma_2}(l_T(x_0, y_0, R_2)),\\
&\ldots\\
&E_{\Sigma_Q}(l_1(x_0, y_0, R_Q)), \ldots, E_{\Sigma_Q}(l_T(x_0, y_0, R_Q))]
\end{aligned}$$

where lj(x0, y0,R) is the location with distance R from (x0, y0) in the direction given by j when the direction is quantized into T values. R = k × S(x0, y0), R1 = R, R2 = 2R, and RQ = Q × R, where S is the characteristic scale map computed in step 1). Therefore, LDFT performs on the same content region even if the image is scaled and the LDFT is scaling invariant. k is a positive number, which is used to adjust the radius of the local region. If k is too large, the LDFT will cover the most part of the image and will not obtain the local effect. If k is too small, the LDFT will cover a very small region; each value of the LDFT vector will be very close. The distinctiveness of the LDFT vector will reduce. We set it as k = 3 in all experiments, which obtains a balance between local effect and distinctiveness. On the other hand, in the watermarking system, k can be set as a secret key, and the receiver who does not know it will not be able to generate accurate watermarking information.

Step 6) Orientation assignment.

By assigning a consistent orientation to the LDFT of each pixel based on local image properties, the LDFT can be represented relative to this orientation and therefore achieves invariance to image rotation.We propose an approach resembling the histogram of oriented gradients (HOG) [39] to gain the consistent orientation. An orientation histogram is formed from convolved orientation maps of sample points within a region around the center pixel (x0, y0). The orientation histogram has H bins covering the 360◦ range of orientations. Note that H = 36 when 36 orientations are considered. Each sample is added to the histogram. We get the histogram of the oth bin with the following equation:

$$\begin{aligned}
&\mathcal{H}_o(x_0, y_0) = \tilde{G}_o^{\Sigma_1}(x_0, y_0)\\
&+ \tilde{G}_o^{\Sigma_1}(l_1(x_0, y_0, R_1)) + \ldots + \tilde{G}_o^{\Sigma_1}(l_T(x_0, y_0, R_1))\\
&+ \tilde{G}_o^{\Sigma_2}(l_1(x_0, y_0, R_2)) + \ldots + \tilde{G}_o^{\Sigma_2}(l_T(x_0, y_0, R_2))\\
&+ \ldots\\
&+ \tilde{G}_o^{\Sigma_Q}(l_1(x_0, y_0, R_Q)) + \ldots + \tilde{G}_o^{\Sigma_Q}(l_T(x_0, y_0, R_Q))
\end{aligned}$$

The maximum peak in the orientation histogram corresponds to dominant direction, denoted by J.

$$J = \arg \max_{o \in [1,H]} \{\mathcal{H}_o(x_0, y_0)\}$$

The initial LDFT ˆD (·) is then recomposed in the clockwise direction from the dominant direction J in each circular layer. Now, the LDFT becomes

$$\begin{aligned}
&D(x_0, y_0) = [E_{\Sigma_1}(x_0, y_0),\\
&E_{\Sigma_1}(l_J(\cdot)), \ldots, E_{\Sigma_1}(l_T(\cdot)), E_{\Sigma_1}(l_1(\cdot)), \ldots, E_{\Sigma_1}(l_{J-1}(\cdot)),\\
&E_{\Sigma_1}(l_J(\cdot)), \ldots, E_{\Sigma_2}(l_T(\cdot)), E_{\Sigma_2}(l_J(\cdot)), \ldots, E_{\Sigma_2}(l_{J-1}(\cdot)),\\
&\ldots\\
&E_{\Sigma_2}(l_J(\cdot)), \ldots, E_{\Sigma_2}(l_T(\cdot)), \ldots, E_{\Sigma_Q}(l_1(\cdot)), \ldots, E_{\Sigma_Q}(l_{J-1}(\cdot))]
\end{aligned}$$

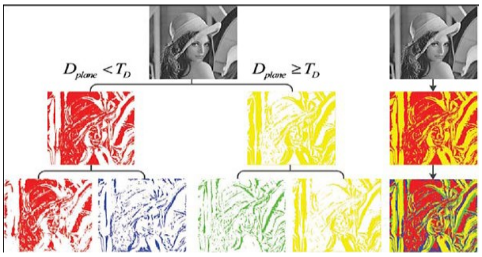
where $E_{\Sigma_q}(l_j(\cdot)) = E_{\Sigma_q}(l_j(x_0, y_0, R_q)),\ j \in [1, T]$, j Є[1,T] and q ∈ [1,Q]. Note that J is the dominant direction, which is the starting direction. Thus, the LDFT consists of 1 + T × Q values, which are extracted from 1 + T × Q locations, respectively.

B. Partitioning LDFT Space Using the BSP Tree

The accurate partitioning of multidimensional feature space is usually difficult due to the large number of feature dimensions and their overlapping distribution in space. Most of existing methods rely on clustering techniques .However, these methods do not resist image cropping attack. The previous method based on k-means clustering is robust against image cropping. However, the watermarking method is semiblind because the centroids of clusters must be sent to the extractor. In this section, we adopt the BSP tree for partitioning the LDFT space. BSP tree construction is a process which takes a subspace and partitions it by any hyperplane that intersects the interior of that subspace. It generates two new subspaces that can be further partitioned by recursive processes. The algorithm to build a BSP tree for LDFT space partitioning consists of three stages.

1) Select a partition plane: Di(·) = TD can be regarded as the ith partition plane, where Di(·) is the ith feature value of the LDFT and i ∈ [1,1 + T × Q]. The partition plane of a space can no longer be used for the subspace. To enhance security, the order of the partition plane can be scrambled by a random sequence with a key Kp. In

order to achieve good results for constructing a balanced and robust BSP tree, where each leaf contains roughly the same number of pixels, we set the threshold TD as the mean value of all Di(·) of pixels in the original image.
2) Partition the set of space by the plane: If the subset is under the partition plane and the value of the feature vector is less than TD, the subset is on the left ramification of the BSP tree; otherwise, the subset is on the right ramification, as shown in Fig 2.



3) Recur with each of the two new subsets: Repeat steps 1) and 2) from left to right until the number of leaf nodes N_{leaf} is equal to the length of watermark sequence N_w.

The BSP is a complete binary tree. The relationship between the number of partition planes N_{pp} and the number of leaf nodes N_{leaf} is $2N_{pp} \geq N_{leaf}$. As shown in Fig. 2, the feature space of the Lena image is partitioned into four subspaces by using the BSP tree, where two feature values of the LDFT D(·) are used for partitioning.

B. Watermark Embedding

After the BSP tree is built according to the length of the watermark sequence N_w, the watermarking sequence W is embedded bit by bit into each leaf node from left to right by using LQIM. The LQIM is a quantization-based data hiding method. Inspired by µ-law concept, the host signal is transformed into logarithmic domain using a compression function. The watermark data are embedded into the transformed signal using uniform quantization, and then, the quantized signal is transformed into the original domain using inverse function. Due to the logarithmic function, smaller step sizes are devoted to smaller amplitudes and vice versa. Compared with UQIM , the LQIM poses perceptual advantages that lead to stronger watermark insertion. For leaf node n, according to the corresponding watermark bit w_n, $w_n \in \{0, 1\}$, each pixel f(x, y) is quantized with a quantizer LQ(·;w) as

$$f_w(x,y) = LQ\left(f(x,y); w_n\right)$$

Concretely, the pixel value f(x, y) must be transformed first by using the following compression function:

$$C(x,y) = \frac{\ln\left(1 + \mu\frac{f(x,y)}{X_s}\right)}{\ln(1+\mu)}, \quad \mu > 0, X_s > 0$$

where µ is a parameter defining the compression level and Xs is the parameter that scales the pixel values of the image. The best Xs value is the value which spreads most of the host signal samples into the range [0, 1]. The transformed signal C(x, y) is then used for data embedding. In this regard, the transformed signal C(x, y) is quantized uniformly by the UQIM watermark embedding method $C_w(x, y) = UQ(C(x, y);wn)$ where UQ(·;w) is a uniform scalar quantizer with a step Δand the quantizer set consists of two quantizers shifted by Δ/2 with respect to each other. Δ is used to control the embedding distortion.

The quantized pixel value is then expanded to obtain the watermarked pixel value as follows:

$$f_w(x,y) = \frac{X_s}{\mu}\left[(1+\mu)^{C_w(x,y)} - 1\right]$$

After every pixel in each leaf node is quantized according to the corresponding watermark bit w_n, the watermark embedding process is completed.

D. Watermark Extraction

The watermark extracting process, similar to watermark embedding, consists of three steps: 1) constructing the feature space with LDFT; 2) building the BSP tree along with partitioning the LDFT space; and 3) extracting the watermark. For each pixel f'(x, y) in leaf node n of the attacked image, determine the embedded watermark bit with the Euclidean distance decoder. Bits 0 and 1 are embedded in the attacked pixel value f'(x, y) using the LQIM embedding method, resulting in f'_0 (x, y) and f'_1 (x, y), respectively. The watermark data can be extracted by the following equation:

$$\hat{w} = \arg\min_{i\in\{0,1\}} \|f'(x,y) - f'_i(x,y)\|^2$$

Where $\hat{w}$ is the extracted watermark data.

When desynchronization or common image-processing attacks occur, even in the same leaf node, some pixels are detected to embed bit 1, and some are 0. Let $\mathrm{Num}_1(n)$ and $\mathrm{Num}_0(n)$ denote the number of pixels hiding bit 1 and bit 0 in leaf node n, respectively. The nth bit of the watermark sequence $\hat{W}$, denoted by ˆ w_n, is extracted as

$$\hat{w}_n = \begin{cases} 1, & \text{if } Num_1(n) \geq Num_0(n) \\ 0, & \text{if } Num_1(n) < Num_0(n) \end{cases}$$

where $\hat{w}$is the extracted watermark data.

E. Attack detection

Watermark can be correctly extracted from the image , even if it has undergone any type of attacks, using the above explained collection of techniques and thus the receiver can ensure the authenticity of the received document(image). For checking the integrity of the watermarked image, the hash is calculated and embedded in the image . The Secure Hash Algorithm-1(SHA-1) is used for the calculation of the hash of the image. It is done by dividing the image into a number of blocks , as SHA-1 processes the input message as 512bit blocks. The message digest is 160 bit. At the receiver side, hash is extracted and compared with recomputed hash to detect the occurrence of attacks.

IV. Conclusion

We have presented in the paper an improved watermarking resynchronization scheme which is robust against local and global desynchronization attacks , and can detect image tampering. Here, Local Daisy Feature Transform and Binary space partitioning tree are used for watermarking in images. We also propose a method which uses Secure Hash Algorithm-1for tampering detection. This watermarking resynchronization scheme is resilient against local and global desynchronization attacks.

Acknowledgment

The authors wish to thank the Management and Principal and Head of the Department(CSE) of Ilahia College of Engineering and Technology for their support and help in completing this work.

References

[1] Huawei Tian, Yao Zhao, Rongrong Ni ,Lunming Qin, and Xuelong Li, , “LDFT based Watermarking resilient to Local Desynchronizationn Attcks,” IEEE Transactions On Cybernetics January 31, 2013.

[2] J. 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.

[3] C. Tang and H. Hang, “A feature-based robust digital image watermarking scheme,” IEEE Trans. Signal Process., vol. 51, no. 4, pp. 950–959, Apr. 2003.

[4] P. Dong, J. G. Brankov, N. P. Galatsanos, Y. Yang, and F. Davoine, “Digital watermarking robust to geometric distortions,” IEEE Trans. Image Process., vol. 14, no. 12, pp. 2140–2150, Dec. 2005.

[5] S. S. Jin and D. Y. Chang, “Image watermarking based on invariant regions of scale-space representation,” IEEE Trans. Signal Process., vol. 54, no. 4, pp. 1537–1549, Apr. 2006.

[6] Y.-T. Wu and F. Y. Shih, “Genetic algorithm based methodology for breaking the steganalytic systems,” IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 36, no. 1, pp. 24–31, Feb. 2006

[7] J. Dugelay, S. Roche, C. Rey, and G. Doërr, “Still-image watermarking robust to local geometric distortions,” IEEE Trans. Image Process., vol. 15, no. 9, pp. 2831–2842, Sep. 2006.

[8] E. Tola, V. Lepetit, and P. Fua, “DAISY: An efficient dense descriptor applied to wide-baseline stereo,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 32, no. 5, pp. 815–830, May 2010.

[9] V. Licks, F. Ourique, R. Jordan, and F. Perez-Gonzalez, “The effect of the random jitter attack on the bit error rate performance of spatial domain image watermarking,” in Proc. Int. Conf. Image Process., Sep. 2003, vol. 3, pp. 455–458.

[10] H. Lee, I. Kang, H. Lee, and Y. Suh, “Evaluation of feature extraction techniques for robust watermarking,” in Proc. 4th Int. Workshop Digit. Watermarking, Sep. 2005, vol. 1, pp. 418–431.

[11] N. K. Kalantari and S. M. Ahadi, “A logarithmic quantization index modulation for perceptually better data hiding,” IEEE Trans. Image Process., vol. 19, no. 6, pp. 1504–1517, Jun. 2010.