PERFORMANCE COMPARISON OF HYBRID WAVELET TRANSFORMS FORMED USING DCT, WALSH, HAAR AND DKT IN WATERMARKING

Dr. H. B. Kekre[1], Dr. Tanuja Sarode[2], Shachi Natu[3]

[1]Senior Professor, Computer Engineering Department, MPSTME, NMIMS University, Mumbai, India
[2]Associate Professor, Computer Engineering Department, TSEC, Mumbai University Mumbai, India
[3]Ph. D. Research Scholar, Computer Engineering Department, MPSTME, NMIMS University, Mumbai, India

ABSTRACT

In this paper a watermarking method using hybrid wavelet transform and SVD is proposed. Hybrid wavelet transform generated from two different orthogonal transforms is applied on host and SVD is applied to watermark. The transforms used for hybrid wavelet transform generation are DCT, Walsh, Haar and DKT. First component transform used in generation of hybrid wavelet transform corresponds to global properties and second component transform corresponds to local properties of an image to which transform are applied. Aim of proposed watermarking method is to study effect of selecting DCT as global/local component transform on robustness. After testing the proposed method against various attacks, using DCT as global component is observed to be robust against compression, resizing using transforms, resizing using grid based interpolation and noise addition attacks. DCT when used as local component is observed to be robust against cropping. It also shows robustness against resizing using transforms, resizing using grid based interpolation and noise addition attacks.

KEYWORDS

Watermarking, Hybrid Wavelet Transform, Singular Value Decomposition, Discrete Kekre Transform

1. INTRODUCTION

Security of digital contents is a major issue when they are transmitted over network using powerful technology like internet. Especially protecting ownership of digital contents so that unauthorised person cannot claim the ownership also known as copyright protection is desired. Inserting information of owner in digital contents to protect copyright popularly known as digital watermarking is adapted. Depending on digital contents to be protected, it can be image watermarking, audio watermarking or video watermarking. Imperceptibility and robustness are the two major requirements of good watermarking algorithm and there is always trade-off between the two. Watermarking methods can be further classified based on how watermark is inserted in host. In case of digital image, pixel values of image can be directly modified to hide the watermark. This is known as spatial domain watermarking. In another type, image is represented in another form using suitable transform and then watermark is inserted in image by modifying these values of transformed image. This type of watermarking is known as transform domain or frequency domain watermarking. Due to high robustness, transform domain watermarking is more popular than spatial domain watermarking. Among transform domain watermarking, various orthogonal transforms, wavelet transforms, singular value decomposition and combination of two or more of them are successfully used. In this paper, an invisible and

DOI:10.5121/ijcsit.2015.7105

robust image watermarking in hybrid wavelet transform domain is proposed. A hybrid wavelet transform to be applied to images is generated by using existing orthogonal transforms like DCT, Walsh, and Haar etc. To increase robustness, hybrid wavelet transform of host is accompanied by singular value decomposition of watermark. Remaining paper is organized as follows: section 2 gives review of literature. Section 3 presents in brief hybrid wavelet transform and singular value decomposition. Section 4 presents proposed watermarking method. Section 5 discusses the performance of proposed method against various attacks. Section 6 presents conclusion of presented work.

2. REVIEW OF LITERATURE

Due to higher robustness, frequency domain watermarking is more popular. Lot of work has been done in transform domain watermarking using DCT [1], [2], [3], wavelet transform [4], [5], [6] singular value decomposition [7], [8] and wavelet packet transform [9]. Methods are also proposed using combination of two or more transforms like DWT-DCT [10], DWT-SVD [11], DCT-SVD [12]; DWT-DCT-SVD [13] Combination of two or more transforms has proved to be more robust than using any single transformation technique.

A. Umaamaheshvari and K. Thanushkodi proposed a watermarking technique based on feature and transform method [14]. Features from cover image are extracted using Harris Laplacian detector. Group of these extracted features forms a primary feature set to embed secret image. Another novel approach of robust watermarking was proposed by Haijun Luo et al [15]. From a host image, sub-images are selected to embed the watermark. In DFT domain of these sub-images watermark is embedded. For restoring the watermark, feature points are extracted using Scale Invariant Fourier Transform. Singular value decomposition (SVD) based technique was proposed by Chih-Chin et al. [16] in which authors explored the D and U components for watermark embedding. Two properties preserved by this technique are namely non-symmetric and one-way. Lagzian et al. proposed a hybrid watermarking scheme [17] with the objective of providing imperceptibility and robustness requirements. The objective was achieved by incorporating two models namely discrete wavelet transform (DWT) and SVD. The watermark was embedded to the elements of singular values of wavelet transformed cover image sub-bands. Li also used DWT and SVD technique for watermarking but in addition, Arnold transform was used to provide security to the watermark [18]. Chang et al. proposed watermarking technique by using redundant discrete wavelet transform (RDWT) instead of DWT and SVD [19]. RDWT was applied to watermark and cover image, and SVD was applied to the LL sub-bands. Another watermarking technique using SVD was proposed by Rastegar et al. [20]. This method has used Finite Radon Transform (FRAT) along with SVD for watermarking. A digital watermarking algorithm for a color watermark embedded into a color host image, based on color space transform and IWT (Integer Wavelet Transform), is proposed by Qingtang Su et al. [21]. According to the Human Visual System peculiarity and quantizing the wavelet coefficient, Encrypted watermark is embedded adaptively into the luminance Y of the YIQ mode in IWT domain. Ying Zhang, Jiqin Wang, Xuebo Chen proposed a watermarking algorithm for color images based on wavelet analysis [22]. The algorithm scrambled the original watermark image in pre-treatment, and used the wavelet transform to process the carrier image and the scrambled watermark image. Then the color watermarked image was embedded into the low-frequency discrete wavelet coefficient of the color carrier image.

3. HYBRID WAVELET TRANSFORM AND SINGULAR VALUE DECOMPOSITION

3.1. Hybrid Wavelet Transform

Kekre et. al proposed an algorithm [23] to generate wavelet transform from two different orthogonal transforms. Being a combination of two transforms, it combines good properties of both the component transforms. On the other hand being a wavelet transform it also provides advantages of wavelet transform. If we have two transform matrices A and B of sizes mxm and nxn respectively, then a hybrid wavelet transform matrix of size mnxmn is generated using the algorithm in [23]. We call A and B as component transform matrices. By varying sizes of these transform matrices; contribution of global and local properties of transform matrix can be varied.

3.2. Singular Value Decomposition

Singular value decomposition is a numerical technique used to diagonalize matrices in numerical analysis. Using singular value decomposition, any real matrix A can be decomposed into a product of three matrices U, S and V as A=USVT, where U and V are orthogonal matrices and S is diagonal matrix. If A is mxn matrix, U is mxm orthonormal matrix whose columns are called as left singular vectors of A and V is nxn orthonormal matrix whose columns are called right singular vectors of A. For m>n, S takes the following form [21]:

$$S=\begin{bmatrix} s1 & 0 & .. & 0 \\ 0 & s2 & .. & 0 \\ . & . & . & . \\ 0 & . & 0 & sn \\ 0 & 0 & .. & 0 \end{bmatrix}$$

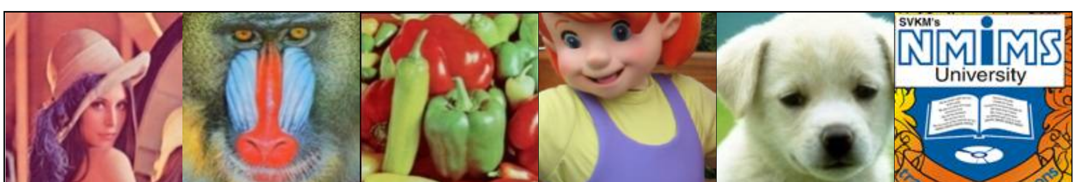
The diagonal elements are listed in descending order, s1≥s2≥….≥sn≥0.

Some properties of SVD which make it useful in image processing are:

- The singular values are unique for a given matrix.
- The rank of matrix A is equal to its nonzero singular values. In many applications, the singular values of a matrix decrease quickly with increasing rank. This property allows us to reduce the noise or compress the matrix data by eliminating the small singular values or the higher ranks [22].
- The singular values of an image have very good stability i.e. when a small perturbation is added to an image; its singular values don't change significantly [23].

4. PROPOSED METHOD

In proposed watermarking method five color images of size 256x256 are used to embed the watermark and a color bitmap image of size 128x128 is used as a watermark. Set of these images is shown in Fig. 1.



(a) Lena | (b) Mandrill | (c) Peppers | (d) Face | (e) Puppy | (f) NMIMS

Figure 1 (a)-(e) Host images and (f) watermark image used for experimental work

Using Kekre's algorithm of wavelet generation [23], a wavelet transform matrix is generated using two different orthogonal transform matrices of different sizes. Based on the size of component transform, numbers of rows in the resultant hybrid wavelet transform matrix contributing to global and local properties of transformed image vary. For example if 256x256 hybrid wavelet matrix is generated using 32x32 DCT matrix and 8x8 Walsh matrix, then according to the algorithm in [23], first 32 rows contribute to global properties and remaining rows which are obtained by shift and rotation contribute to local properties of image. Thus in the proposed method, hybrid wavelet transform is generated from DCT as global component combined with Walsh, Haar and DKT with size combinations (64, 4), (32, 8), (16, 16), (8, 32) and (4, 64) for each. With same size combinations but DCT as second component transform i.e. local component and other transforms as global transforms, results are studied and analysed.

4.1. Embedding Procedure

1. Generate 256x256 DCT-Walsh hybrid wavelet transform using DCT matrix and Walsh matrix with above mentioned size combinations for host.
2. Apply Singular Value Decomposition (SVD) to watermark image. Due to high energy compaction property of SVD, only first few singular values are sufficient to represent the image and can be used to embed in the host image. After trying different values, first 30 singular values are found to be sufficient without losing visual quality of an image. Thus first 30 values are selected for embedding.
3. Apply DCT-Walsh hybrid wavelet transform to host image columns. This column transform leads to energy compaction of an image in upper rows containing low frequency components. Since we need middle hybrid wavelet coefficients, middle rows are selected to embed the watermark. After exhaustive testing, rows 101-130 are found suitable as mid-frequency coefficients to embed watermark as embedding in these rows is robust against maximum attacks than other rows.
4. Hybrid wavelet coefficients from selected mid-frequency band are sorted in the decreasing order of their energy.
5. By using the highest energy coefficient and first singular value of watermark, weight factor is calculated. Using this weight factor, all singular values are scaled down.
6. First scaled singular value is used to replace highest energy coefficient. Second scaled singular value is used to replace the wavelet coefficient that is just higher than it.
7. Remaining singular values are placed consecutive to second singular value. Index values where these singular values are replaced are recorded. This helps to minimize the energy gap between the host wavelet coefficients and scaled singular values thereby reducing the distortion in watermarked image.
8. Inverse hybrid wavelet transform is applied to get watermarked image.

4.2. Extraction process

1. Apply hybrid wavelet transform to watermarked image.
2. Extract the mid frequency coefficients and sort them in the decreasing order of their energy.
3. From the index values recorded in embedding procedure, singular values are obtained.
4. These singular values are scaled up using the weight factor computed in embedding process.
5. Inverse singular value decomposition is applied to get watermark.
6. Average of absolute pixel difference between embedded and extracted watermark (Mean Absolute Error i.e. MAE) is calculated to measure the robustness.
7. Embedding and extraction steps are repeated using row hybrid wavelet transform with DCT as global and then local component.

Further, different attacks like compression, cropping, noise addition and resizing are performed on watermarked image. Extraction procedure is applied on attacked watermarked image to recover watermark from it. Performance analysis of proposed method when sinusoidal transform DCT used as global component transform and local component transform is given in next section.

5. PERFORMANCE ANALYSIS OF PROPOSED METHOD

5.1. Compression attack

Watermarked images are subjected to compression using different techniques namely using transforms, JPEG compression and using Vector Quantization (VQ). DCT, DST, Walsh, Haar and DCT wavelet are the transforms used to compress watermarked image. Since embedding is done by applying column transform to host image, compression of watermarked image is also performed by applying column transform and then by eliminating high frequency coefficients to get compression ratio 1.142. For JPEG compression, quality factor 100 is used. For VQ, Kekre's Fast Codebook Generation (KFCG) algorithm [24] is used and image is compressed by generating codebook of size 256.

5.1.1 DCT-Walsh and Walsh-DCT hybrid wavelet transform

Fig. 2 shows result images for compression attack using DCT when DCT-Walsh and Walsh-DCT obtained from (16, 16) size combinations are used to embed watermark.

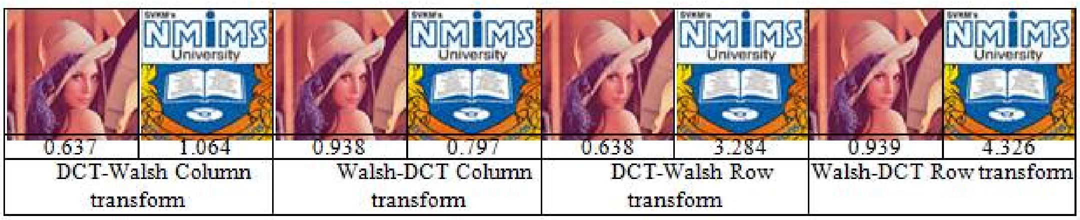


Figure 2 watermarked image Lena after compressing using DCT and watermark extracted from it when embedding is done using DCT-Walsh and Walsh-DCT column and row transforms obtained using (16,16) size component transforms.

From Fig. 2 it is observed that when DCT-Walsh hybrid wavelet is used either in column or row form, extracted watermark closely matches with embedded watermark. Use of Walsh-DCT hybrid wavelet in column form gives slightly better quality of extracted watermark. Walsh-DCT when applied in row form gives comparatively higher MAE between embedded and extracted watermark. In both, DCT-Walsh and Walsh-DCT, column or row transform does not cause much difference in quality of extracted watermark as well as imperceptibility of watermarked image. Since five host images are used, performance of proposed method against compression attack is judged by calculating average of MAE between embedded and extracted watermark from five host images.

Table 1 below shows average of MAE between embedded and extracted watermark against compression attack when embedding procedure is carried out using column version of DCT-Walsh and Walsh-DCT hybrid wavelet transform of different size combinations of DCT and Walsh transforms.

Table 1 Average MAE between embedded and extracted watermark against compression attack using column DCT-Walsh hybrid wavelet and column Walsh-DCT hybrid wavelet obtained from different sizes of component transforms

Compression Type	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT
	64by4	64by4	32by8	32by8	16by16	16by16	8by32	8by32	4by64	4by64
DCT	1.997	12.661	2.750	9.875	1.652	6.958	1.440	5.150	1.416	3.310
DST	2.088	12.941	2.606	9.772	1.700	6.928	1.497	5.135	1.439	3.299
Walsh	0.000	17.373	0.000	22.600	0.000	16.840	0.000	12.630	0.000	11.966
Haar	0.000	17.114	0.000	17.214	0.000	36.397	0.000	49.284	0.000	17.821
JPEG	61.032	60.624	65.282	64.548	63.002	69.599	63.491	67.406	64.984	73.487
VQ compression	51.395	47.701	47.925	47.291	51.910	58.214	54.687	61.255	54.166	59.081
DCT wavelet	47.142	34.372	56.917	31.713	54.575	0.000	56.897	50.222	51.645	42.518

From Table 1 evident observation that can be made is irrespective of different sizes of component transforms used to generate DCT-Walsh or Walsh-DCT hybrid wavelet, DCT when used as global component transform i.e. as DCT-Walsh, gives better robustness than Walsh-DCT. The fluctuation in MAE is also very small. Whereas, for Walsh-DCT transform used for embedding, continuous decrease in MAE is observed in compression using DCT, DST, Walsh and Haar as size of local transform is increased. This means when we focus more on local properties with smaller resolution, it gives better robustness against compression attack. This does not hold true for compression using DCT wavelet, JPEG compression and VQ compression attack. There continuous fluctuations and higher MAE values are observed except Walsh-DCT wavelet giving zero MAE against compression using DCT wavelet.

Table 2 shows performance of row DCT-Walsh and row Walsh-DCT hybrid wavelet against compression attack.

Table 2 Average MAE between embedded and extracted watermark against compression attack using row DCT-Walsh hybrid wavelet and column Walsh-DCT hybrid wavelet obtained from different sizes of component transforms

Compression Type	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT
	64by4	64by4	32by8	32by8	16by16	16by16	8by32	8by32	4by64	4by64
DCT	1.064	12.259	1.591	9.029	1.282	6.485	1.342	4.637	1.215	2.960
DST	1.048	12.695	1.714	8.999	1.393	6.509	1.417	4.655	1.270	3.035
Walsh	0.000	15.259	0.000	20.532	0.000	16.707	0.000	13.574	0.000	12.528
M Haar	0.000	20.792	0.000	32.886	0.000	27.361	0.000	24.698	0.000	33.989
JPEG	62.116	62.629	64.931	69.111	65.212	69.099	65.047	71.109	67.027	76.094
VQ compression	40.570	40.151	47.173	49.102	55.267	59.833	57.135	61.159	58.615	62.162
DCT wavelet	43.366	27.538	54.589	26.201	58.872	0.000	54.823	28.597	58.150	51.150

Similar to results of column hybrid wavelet transforms, row DCT-Walsh transform shows better robustness than row Walsh-DCT hybrid wavelet transform when compression is done using DCT, DST, Walsh and Haar. For JPEG compression and VQ compression both the transforms show higher values of MAE but DCT-Walsh shows it slightly better than Walsh-DCT wavelet. For compression using DCT wavelet, Walsh-DCT wavelet shows better robustness than DCT-Walsh wavelet.

5.1.2 DCT-Haar and Haar-DCT hybrid wavelet transform

Table 3 shows performance comparison of proposed method against compression attack when DCT is used as global and then local transform along with Haar as another component transform.

This combination results in DCT-Haar and Haar-DCT hybrid wavelet transforms which are applied column-wise and row-wise on host to embed and extract watermark.

Table 3 Average MAE between embedded and extracted watermark against compression attack using column DCT-Haar hybrid wavelet and column Haar-DCT hybrid wavelet obtained from different sizes of component transforms

Compression type	DCT-Haar 64by4	Haar-DCT 64by4	DCT-Haar 32by8	Haar-DCT 32by8	DCT-Haar 16by16	Haar-DCT 16by16	DCT-Haar 8by32	Haar-DCT 8by32	DCT-Haar 4by64	Haar-DCT 4by64
DCT	4.036	12.661	4.283	9.875	2.742	6.958	2.786	5.150	3.206	3.310
DST	3.825	12.941	4.288	9.772	2.881	6.928	2.878	5.135	3.438	3.299
Walsh	8.337	17.373	4.855	22.600	3.557	16.840	2.682	12.630	4.291	11.966
Haar	**0.000**	17.114	**0.000**	17.214	**0.000**	36.397	**0.000**	49.284	**0.000**	17.821
JPEG	59.828	60.624	56.804	64.548	58.148	69.599	58.843	67.406	57.946	73.487
VQ compression	50.400	47.701	40.194	47.291	39.260	58.226	41.140	61.256	45.926	59.080
DCT wavelet	43.511	34.372	47.070	31.713	44.043	**0.000**	43.458	50.222	45.817	42.518

From Table 3 it is observed that for compression using DCT and DST, as the contribution of DCT as local transform increases and that of Haar as global component decreases (i.e. Haar-DCT), robustness improves. DCT when used as global component transform along with Haar as local, performance is consistently better than Haar-DCT column hybrid wavelet. For Walsh and Haar based compression attack also, DCT-Haar shows higher robustness when DCT-Haar wavelet is used. Especially for compression using Haar transform, any size combination for DCT-Haar gives excellent robustness with zero MAE. For JPEG compression though MAE values are high, they are smaller for DCT-Haar column hybrid wavelet transform as compared to Haar-DCT. For compression using DCT wavelet, Haar-DCT wavelet proves better in robustness. Haar-DCT column wavelet generated using 16x16 size Haar and DCT gives exceptionally withstands against DCT wavelet based compression. For VQ based compression performance of both DCT-Haar and Haar-DCT keeps on fluctuating.

Table 4 shows performance comparison of DCT-Haar and Haar-DCT row wavelet transforms against compression attack.

Compression Type	DCT-Haar 64by4	Haar-DCT 64by4	DCT-Haar 32by8	Haar-DCT 32by8	DCT-Haar 16by16	Haar-DCT 16by16	DCT-Haar 8by32	Haar-DCT 8by32	DCT-Haar 4by64	Haar-DCT 4by64
DCT	1.843	12.259	1.941	9.029	2.081	6.485	3.091	4.637	3.376	2.960
DST	1.806	12.695	1.865	8.999	2.265	6.509	3.340	4.655	3.464	3.035
Walsh	2.511	15.259	1.113	20.532	2.311	16.707	4.284	13.574	2.922	12.528
M Haar	0.000	20.792	0.000	32.886	0.000	27.361	0.000	24.698	0.000	33.989
JPEG	62.003	62.629	61.015	69.111	59.705	69.099	60.675	71.109	60.250	76.094
VQ compression	40.709	40.151	43.542	49.102	39.278	59.833	41.548	61.159	39.806	62.162
DCT wavelet	43.776	27.538	42.423	26.201	41.510	0.000	46.681	28.597	47.210	51.150

Observations noted from Table 4 are similar to that of Table 3. For compression attack using DCT, DST, Walsh and Haar transform, DCT-Haar better sustains than Haar-DCT. Against JPEG compression poor resistance is observed by both DCT-Haar and Haar-DCT row wavelet transform. However, as size of DCT as global component transform is reduced, this MAE decreases. In contrast, as size of DCT as local transform is increased, MAE gradually increases. For compression using DCT wavelet, Haar-DCT row wavelet better withstands than DCT-Haar and shows excellent robustness with zero MAE at size combination (16, 16).

5.1.3. DCT-DKT and DKT-DCT hybrid wavelet transform

Table 5 shows performance of DCT-DKT and DKT-DCT column hybrid wavelet transform with their various size combinations against compression attack performed using transforms like DCT, DST, Walsh, Haar and DCT wavelet and compression using JPEG and VQ.

Table 5 Average MAE between embedded and extracted watermark against compression attack using column DCT-DKT hybrid wavelet and column DKT-DCT hybrid wavelet obtained from different sizes of component transforms

Compression Type	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT
	64by4	64by4	32by8	32by8	16by16	16by16	8by32	8by32	4by64	4by64
DCT	23.056	12.661	24.493	9.875	18.340	6.958	20.414	5.150	19.610	3.310
DST	23.284	12.941	24.534	9.772	18.223	6.928	20.576	5.135	19.641	3.299
Walsh	32.510	17.373	28.812	22.600	26.654	16.840	24.644	12.630	23.614	11.966
Haar	65.177	17.114	34.061	17.214	30.963	36.397	28.771	49.284	31.744	17.821
JPEG	74.670	60.624	68.299	64.548	68.723	69.599	67.241	67.406	69.165	73.487
VQ compression	58.687	47.701	44.104	47.291	44.389	58.196	41.091	61.241	42.255	59.077
DCT wavelet	75.914	34.372	72.287	31.713	64.838	0.000	63.436	50.222	62.735	42.518

From the results summarized in Table 5, it is clear that when DCT is used as local component transform with DKT as global one, gives better resistance against compression attack (using various transforms) than using DCT as global component transform with DKT. Also as resolution of local properties of an image is reduced, better robustness is observed. For compression using vector quantization, though MAE values between embedded and extracted watermark are higher, they are better for DCT-DKT column wavelet transform for all possible size combinations except (64,4). For JPEG compression, as we go on increasing contribution of local component transform (either DCT or DKT), DCT-DKT shows marginally better performance over DKT-DCT column wavelet.

Table 6 shows the results of row hybrid wavelet transform DCT-DKT and DKT-DCT against compression attack.

Table 6 Average MAE between embedded and extracted watermark against compression attack using row DCT-DKT hybrid wavelet and row DKT-DCT hybrid wavelet obtained from different sizes of component transforms

Compression Type	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT
	64by4	64by4	32by8	32by8	16by16	16by16	8by32	8by32	4by64	4by64
DCT	25.992	12.259	19.532	9.029	19.257	6.485	19.276	4.637	20.752	2.960
DST	26.053	12.695	19.332	8.999	19.197	6.509	19.466	4.655	20.781	3.035
Walsh	37.167	15.259	27.359	20.532	25.433	16.707	26.464	13.574	25.294	12.528
M Haar	61.405	20.792	42.377	32.886	28.777	27.361	26.758	24.698	31.084	33.989
JPEG	71.516	62.629	70.738	69.111	70.477	69.099	67.930	71.109	70.081	76.094
VQ compression	47.051	40.150	46.973	49.102	44.323	59.907	41.921	61.161	46.289	62.163
DCT wavelet	76.033	27.538	67.332	26.201	64.917	0.000	62.254	28.597	65.282	51.150

Observations for performance of row DCT-DKT and DKT-DCT wavelet are similar to that of column wavelet transform. In both the cases, DKT-DCT transform obtained from (16, 16) size combinations of DKT and DCT gives zero MAE against compression using DCT wavelet transform.

5.2. Cropping attack

Watermarked image is cropped at different regions and with different amount of information. From watermarked image, total 32x32 size portion is cropped once at centre and then same amount of information is cropped by cutting four squares of size 16x16 each at corners of an image. Also 32x32 size squares are cropped at four corners of image which results in total 64x64 pixels removed from an image.

5.2.1. DCT-Walsh and Walsh-DCT hybrid wavelet transform

Fig. 3 below shows watermarked image Lena when 16x16 size squares are cut from it at corners and recovered watermark from such image.

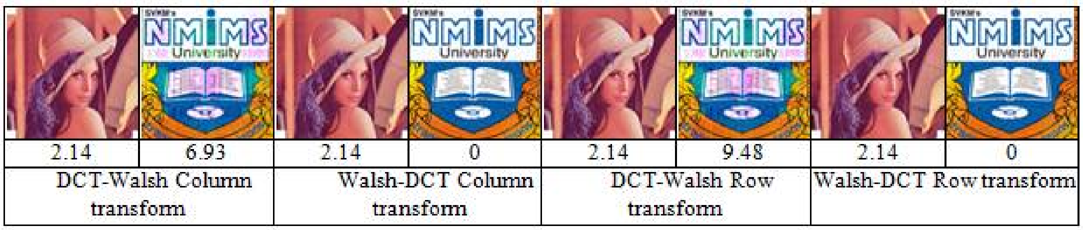


Figure 3 watermarked image Lena after cropping 16x16 portions at corners and watermark extracted from it when embedding is done using DCT-Walsh and Walsh-DCT column and row transforms obtained using (16,16) size component transforms.

Table 7 and Table 8 show comparison of MAE between embedded and extracted watermark from cropped watermarked images where DCT-Walsh and Walsh-DCT are used for embedding watermark.

Table 7 Comparison of MAE between embedded and extracted watermark against cropping attack using DCT-Walsh and Walsh-DCT hybrid wavelet column transform

Cropping Type	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT
	64by4	64by4	32by8	32by8	16by16	16by16	8by32	8by32	4by64	4by64
16x16 crop	18.870	15.214	13.482	12.483	10.177	14.470	5.418	13.657	15.774	32.891
32x32 crop	46.527	31.046	16.225	25.532	26.448	37.386	6.893	7.581	30.197	56.120
32x32crop centre	0.000	0.000	14.214	24.572	12.256	24.271	6.534	24.166	14.662	35.549

From Table 7 it can be seen that when (64,4) and (32,8)size combination is used to generate DCT-Walsh and Walsh-DCT hybrid wavelet, DCT as local component performs better for cropping 16x16 at corners. As we go on increasing size of local component transform further and reducing global component transform, DCT as global component gives significantly better robustness over Walsh-DCT hybrid wavelet transform. For cropping at centre DCT as global component in DCT-Walsh gives consistently better robustness over Walsh-DCT hybrid column wavelet.

Table 8 Comparison of MAE between embedded and extracted watermark against cropping attack using DCT-Walsh and Walsh-DCT hybrid wavelet row transform

Cropping Type	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT
	64by4	64by4	32by8	32by8	16by16	16by16	8by32	8by32	4by64	4by64
16x16 crop	6.199	3.059	5.275	1.835	4.049	2.692	7.439	25.821	14.327	33.239
32x32 crop	21.525	19.618	31.544	16.429	13.518	25.900	15.550	7.011	35.988	63.099
32x32 crop centre	0.000	0.000	2.549	3.310	4.973	3.441	26.627	21.118	21.477	45.252

For row transform, DCT-Walsh gives better robustness when DCT size is taken 8x8 or 4x4 otherwise DCT as local component transform is better in robustness when cropping is done at corners. For 32x32 cropping at corners, irregular fluctuations in MAE values are observed. Walsh-DCT with size combination (8, 32) gives smallest MAE in this case. Similar fluctuations are observed from cropping at centre and both DCT-Walsh and Walsh-DCT with size (64, 4) give equally strong robustness with zero MAE.

5.2.2. DCT-Haar and Haar-DCT hybrid wavelet transform

Table 9 shows MAE between embedded and extracted watermark after performing cropping attack when DCT-Haar and Haar-DCT column wavelet transforms are used for inserting watermark into host.

Table 9 Comparison of MAE between embedded and extracted watermark against cropping attack using DCT-Haar and Haar-DCT hybrid wavelet column transform

Cropping Type	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT
	64by4	64by4	32by8	32by8	16by16	16by16	8by32	8by32	4by64	4by64
16x16 crop	17.323	15.214	2.099	12.483	0.293	14.470	4.459	13.657	1.529	32.891
32x32 crop	24.187	31.046	6.737	25.532	19.375	37.386	12.159	7.581	5.606	56.120
32x32crop centre	0.000	0.000	1.636	24.572	0.244	24.271	5.816	24.166	3.557	35.549

Table 9 shows that DCT-Haar shows very good robustness against 16x16 cropping attack when it is generated using (32,8), (16,16), (8,32) and (4,64). Robustness shown by Haar-DCT for the same attack is also good but not as strong as DCT-Haar. For 32x32 cropping at corners, DCT-Haar column wavelet consistently shows better robustness over Haar-DCT column wavelet transform. Haar-DCT column wavelet shows better robustness than DCT-Haar only for (8, 32) size combination. For 32x32 cropping at centre, DCT-Haar exceptionally performs well with all size combinations over Haar-DCT column wavelet transform.

Table 10 Comparison of MAE between embedded and extracted watermark against cropping attack using DCT-Haar and Haar-DCT hybrid wavelet row transform

Cropping Type	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT
	64by4	64by4	32by8	32by8	16by16	16by16	8by32	8by32	4by64	4by64
16x16 crop	4.049	3.059	4.499	1.835	5.037	2.692	5.230	25.821	4.270	33.239
32x32 crop	19.423	19.618	8.651	16.429	21.545	25.900	12.046	7.011	18.277	63.099
32x32crop centre	0.000	0.000	3.253	3.310	2.984	3.441	1.571	21.118	2.384	45.252

As can be seen from Table 10, DCT-Haar row wavelet transform is consistently giving very good robustness against 16x16 cropping attack. Haar-DCT row wavelet also follows this trend except for size combinations (8, 32) and (4, 64). Against cropping 32x32 at corners, initially DCT-Haar and Haar-DCT row wavelet perform equally well. Later the performance gap between the two is significant with DCT-Haar showing better robustness. For cropping 32x32 at centre, also DC-Haar and Haar-DCT perform equally well for size combinations (64, 4), (32, 8) and (16, 16). Later, DCT-Haar maintains the strong robustness but Haar-DCT shows reduced robustness.

5.2.3. DCT-DKT and DKT-DCT hybrid wavelet transform

Table 11 and Table 12 show performance of column and row wavelet transforms respectively generated using DCT and DKT against cropping attack.

Table 11 Comparison of MAE between embedded and extracted watermark against cropping attack using DCT-DKT and DKT-DCT hybrid wavelet column transform

Cropping Type	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT
	64by4	64by4	32by8	32by8	16by16	16by16	8by32	8by32	4by64	4by64
16x16 crop	16.105	15.214	3.494	12.483	3.955	14.470	29.290	13.657	12.289	32.891
32x32 crop	25.121	31.046	6.803	25.532	39.768	37.386	2.115	7.581	199.689	56.120
32x32crop centre	0.000	0.000	13.725	24.572	3.316	24.271	97.620	24.166	25.098	35.549

From Table 11 it is noted that for 16x16 cropping attack, performance of DCT-DKT and DKT-DCT is widely fluctuating. DCT-DKT with size combination of (32, 8), and (16, 16) give very good robustness. Against cropping 32x32 at corners, DCT-DKT with size combination (32, 8) and (8, 32) gives noticeable good robustness. Against 32x32 cropping at centre, DCT-DKT and DKT-DCT with combination (64, 4) show excellent robustness with zero MAE. For rest of the size combinations, this performance widely fluctuates.

Table 12 Comparison of MAE between embedded and extracted watermark against cropping attack using DCT-DKT and DKT-DCT hybrid wavelet row transform

Cropping Type	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT
	64by4	64by4	32by8	32by8	16by16	16by16	8by32	8by32	4by64	4by64
16x16 crop	4.835	3.059	1.892	1.835	14.207	2.692	20.647	25.821	50.308	33.239
32x32 crop	10.123	19.618	6.368	16.429	18.353	25.900	5.531	7.011	202.193	63.099
32x32cropcenter	0.000	0.000	14.125	3.310	14.940	3.441	129.946	21.118	31.970	45.252

Wide fluctuations observed in the column DCT-DKT and DKT-DCT are now not observed in row transform. Consistently good performance against cropping 16x16 attacks and cropping 32x32 attack (except for the size combination (4, 64)) is given by DCT-DKT row transform. For cropping at centre, DKT-DCT i.e. DCT as local component transform gives better robustness.

5.3. Noise addition attack

Binary distributed run length noise and Gaussian distributed run length noise are two types of noises added to watermarked images. Among them binary distributed run length noise is added with different run length.

5.3.1. DCT-Walsh and Walsh-DCT Hybrid wavelet transform

Fig. 4 shows watermarked images after adding Gaussian distributed run length noise and watermark extracted from it.

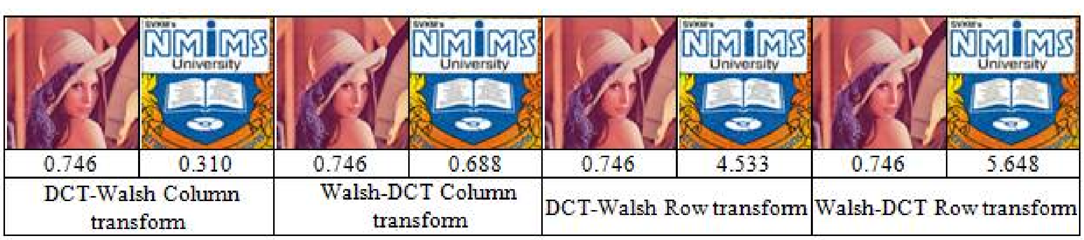


Figure 4 watermarked image Lena after adding Gaussian distributed run length noise and watermark extracted from it when embedding is done using DCT-Walsh and Walsh-DCT column and row transforms obtained using (16,16) size component transforms.

From Fig. 4 it can be observed that for Lena image, column transform using DCT as global/local component transform along with Walsh gives better robustness than row transform when Gaussian distributed run length noise is added to watermarked Lena. In both column and row transforms, DCT-Walsh gives marginally better robustness than Walsh-DCT. Average MAE values against different types of noises added to watermarked images when column hybrid wavelet transforms DCT-Walsh and Walsh-DCT are generated using different sizes of DCT and Walsh are given in Table 13.

In the table, figures in bracket indicate run length.

Table 13 Comparison of MAE between embedded and extracted watermark against noise addition attack using DCT-Walsh and Walsh-DCT hybrid wavelet column transform

Noise Type	**DCT-Walsh**	**Walsh-DCT**	**DCT-Walsh**	**Walsh-DCT**	**DCT-Walsh**	**Walsh-DCT**	**DCT-Walsh**	**Walsh-DCT**	**DCT-Walsh**	**Walsh-DCT**
	64by4	**64by4**	**32by8**	**32by8**	**16by16**	**16by16**	**8by32**	**8by32**	**4by64**	**4by64**
BRLN (1to 10)	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
BRLN (5to 50)	7.221	7.448	6.358	7.621	6.777	8.277	7.421	8.876	8.011	9.640
BRLN (10to100)	7.168	7.459	7.029	7.067	6.283	8.191	7.045	8.414	8.337	10.800
GRLN	0.585	0.560	0.388	0.534	0.915	0.728	1.116	0.908	1.221	0.941

From Table 13, it can be concluded that for smaller run length 1 to 10 of binary distributed run length noise, irrespective of column or row transform and irrespective of whether DCT is used as global or local component transform, proposed method gives highest robustness with zero MAE. For increased run length, for all component sizes, DCT-Walsh gives slightly better robustness than Walsh-DCT hybrid wavelet transform. For Gaussian distributed run length noise, DCT-Walsh as well as Walsh-DCT gives very good robustness though the MAE values are quiet fluctuating.

Table 14 summarizes performance of row versions of DCT-Walsh and Walsh-DCT against noise addition attack.

Table 14 Comparison of MAE between embedded and extracted watermark against noise addition attack using DCT-Walsh and Walsh-DCT hybrid wavelet row transform

Noise Type	**DCT-Walsh**	**Walsh-DCT**	**DCT-Walsh**	**Walsh-DCT**	**DCT-Walsh**	**Walsh-DCT**	**DCT-Walsh**	**Walsh-DCT**	**DCT-Walsh**	**Walsh-DCT**
	64by4	**64by4**	**32by8**	**32by8**	**16by16**	**16by16**	**8by32**	**8by32**	**4by64**	**4by64**
BRLN	3.227	4.236	4.379	3.294	4.897	4.598	4.395	3.800	5.897	5.004
BRLN (5to50)	0.819	1.716	1.411	1.484	2.134	1.753	2.567	2.260	2.847	2.401
BRLN (10 to 100)	1.022	0.835	1.069	0.942	1.256	1.284	1.887	1.485	2.350	1.647
GRLN	5.104	5.089	6.268	6.036	6.913	7.163	7.289	8.123	7.672	8.930

From Table 6, it can be observed that for smaller run length i.e. 1 to 10 of binary distributed run length noise, row wavelet transform give higher MAE than column transform of DCT-Walsh and Walsh-DCT hybrid wavelet transform. Also DCT-Walsh and Walsh-DCT give more or less same robustness. For increased run length, row wavelet transform performs better than column wavelet transform and DCT as global or local component transform shows slight fluctuations in robustness. For Gaussian distributed run length noise robustness shown by DCT-Walsh and Walsh-DCT row wavelet transforms are very good and not much different in robustness. However robustness of column wavelet transform is still better than row hybrid wavelet versions of DCT-Walsh and Walsh-DCT.

5.3.2. DCT-Haar and Haar-DCT hybrid wavelet transform

Table 15 shows performance comparison of DCT-Haar and Haar-DCT column wavelet transform generated from different size combinations of DCT and Haar against noise addition attack.

Table 15 Comparison of MAE between embedded and extracted watermark against noise addition attack using DCT-Haar and Haar-DCT hybrid wavelet column transform

Noise Type	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT
	64by4	64by4	32by8	32by8	16by16	16by16	8by32	8by32	4by64	4by64
BRLN	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
BRLN (5to50)	7.746	7.448	5.541	7.673	5.260	8.197	5.503	9.788	5.921	10.137
BRLN (10 to 100)	7.802	7.459	5.747	7.220	5.089	8.258	5.349	11.281	6.005	10.500
GRLN	0.217	0.560	0.370	0.534	0.493	0.728	0.260	0.908	0.388	0.941

From table 15, following observations are noted. For run length 1 to 10 of binary distributed run length noise, DCT-Haar and Haar-DCT column wavelet transforms both perform exceptionally well irrespective of size combinations used to generate them. For further increased run length 5 to 50 and 10 to 100, both show very good robustness except that Haar-DCT gives slightly higher MAE values. For Gaussian distributed run length noise also both transforms show excellent robustness where MAE given by Haar-DCT is negligibly higher than DCT-Haar.

Table 16 Comparison of MAE between embedded and extracted watermark against noise addition attack using DCT-Haar and Haar-DCT hybrid wavelet row transform

Noise Type	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT
	64by4	64by4	32by8	32by8	16by16	16by16	8by32	8by32	4by64	4by64
BRLN	2.802	4.236	4.820	3.294	3.591	4.598	3.643	3.800	3.974	5.004
BRLN (5to50)	1.504	1.716	1.953	1.484	1.105	1.753	1.617	2.260	1.066	2.401
BRLN (10 to 100)	0.839	0.835	1.262	0.942	0.735	1.284	0.659	1.485	0.487	1.647
GRLN	5.139	5.089	5.797	6.036	4.879	7.163	5.189	8.123	5.103	8.930

As can be seen from Table 14, in row version of DCT-Haar and Haar-DCT wavelet transforms, very good robustness is observed for all types of noises and every possible size combination explored in proposed method. When compared to column version, performance against binary distributed run length noise with run length 5 to 50 and 10 to 100 is improved while performance against Gaussian distributed run length noise and binary distributed run length noise with run length 1 to 10 shows small increase in MAE.

5.3.3. DCT-DKT and DKT-DCT hybrid wavelet transform.

Table 17 shows the performance of DCT-DKT and DKT-DCT column transforms against noise addition attack.

Table 17 Comparison of MAE between embedded and extracted watermark against noise addition attack using DCT-DKT and DKT-DCT hybrid wavelet column transform

Noise Type	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT
	64by4	64by4	32by8	32by8	16by16	16by16	8by32	8by32	4by64	4by64
BRLN	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
BRLN (5to50)	7.857	7.363	5.638	7.527	5.335	8.337	4.859	10.228	5.254	9.259
BRLN(10to 100)	8.343	7.572	5.567	7.180	5.314	8.828	4.945	10.235	5.173	10.967
GRLN	0.082	0.560	0.287	0.534	0.720	0.728	1.353	0.908	1.914	0.941

As observed in Table 17, for binary distributed run length noise (run length noise 1 to 10) and Gaussian distributed run length noise, both DCT-DKT and DKT-DCT show excellent robustness irrespective of size combinations used to generate wavelet transforms. For binary distributed run length noise of run length 5 to 50 and 10 to 100, DCT-DKT column wavelet shows equally well or superior performance over DKT-DCT column wavelet transform.

Table 18 shows performance comparison of DCT-DKT and DKT-DCT row wavelet transform against noise addition attack.

Table 18 Comparison of MAE between embedded and extracted watermark against noise addition attack using DCT-DKT and DKT-DCT hybrid wavelet row transform

Noise Type	**DCT-DKT**	**DKT-DCT**	**DCT-DKT**	**DKT-DCT**	**DCT-DKT**	**DKT-DCT**	**DCT-DKT**	**DKT-DCT**	**DCT-DKT**	**DKT-DCT**
	64by4	**64by4**	**32by8**	**32by8**	**16by16**	**16by16**	**8by32**	**8by32**	**4by64**	**4by64**
BRLN	5.601	4.236	5.138	4.027	4.246	4.598	4.794	4.226	5.636	5.004
BRLN (5to50)	2.003	1.716	1.592	1.789	2.690	1.753	3.475	2.044	4.553	2.401
BRLN (10 to 100)	0.542	0.835	1.305	0.722	1.369	1.284	2.275	1.411	3.224	1.647
GRLN	7.036	5.089	5.912	6.036	5.206	7.163	5.494	8.123	5.576	8.930

For all types of noise attacks, DCT-DKT and DKT-DCT row wavelet transforms show very good robustness. Majority of the times DKT-DCT is marginally better than DCT-DKT row wavelet transform.

5.4. Resizing attack:

Watermarked images are subjected to resizing attack by enlarging them to twice of its original size and then reducing them back to original size. For doing this three approaches namely grid based resizing [25], transform based image zooming [26] and bicubic interpolation are used.

5.4.1. DCT-Walsh and Walsh-DCT Hybrid wavelet transform

Fig. 5 shows resized watermarked image Lena using bicubic interpolation and watermark extracted from it when DCT-Walsh and Walsh-DCT column/row hybrid wavelet transforms are used to embed watermark.

1.25	35.10	1.24	23.306	1.25	29.14	1.24	18.813
DCT-Walsh Column transform		Walsh-DCT Column transform		DCT-Walsh Row transform		Walsh-DCT Row transform	

Fig. 5 watermarked image Lena after resizing using bicubic interpolation and watermark extracted from it when embedding is done using DCT-Walsh and Walsh-DCT column and row transforms obtained using (16,16) size component transforms.

From fig. 5 it is noted that among column and row transforms, row versions of DCT-Walsh and Walsh DCT give better quality extracted watermark than column versions. When compared between DCT-Walsh and Walsh-DCT, Walsh-DCT gives better robustness in both column and row versions. Table 19 shows average MAE values between embedded and extracted watermark against various types of resizing attacks using DCT-Walsh and Walsh-DCT column hybrid wavelet transforms obtained using different sizes of component transforms DCT and Walsh.

Table 19 Comparison of MAE between embedded and extracted watermark against resizing attack using DCT-Walsh and Walsh-DCT hybrid wavelet column transform

Resize Type	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT
	64by4	**64by4**	**32by8**	**32by8**	**16by16**	**16by16**	**8by32**	**8by32**	**4by64**	**4by64**
Resize2	30.933	32.038	29.987	29.494	29.675	27.028	29.934	28.178	29.524	28.875
DFT_resize2	1.154	1.755	1.307	1.798	1.206	1.566	1.043	1.536	1.316	1.793
Grid-resize	4.272	2.968	5.739	3.365	8.105	4.398	11.381	5.305	16.136	8.566

For resizing using bicubic interpolation, performance of DCT-Walsh and Walsh-DCT show fluctuations as size of global and local component transforms is changed. Both these hybrid wavelet transforms show more or less similar performance against resizing using bicubic interpolation. Among transform based resizing, very small MAE between embedded and extracted watermark is observed using DFT for resizing. For DCT-Walsh column wavelet this error is smaller than Walsh-DCT column wavelet. For other transforms like DCT, DST, Haar, Hartley used for resizing, MAE is zero thus showing strong robustness. For grid based resizing also DCT when used as local transform gives very good robustness. Though MAE between embedded and extracted watermark are higher for DCT-Walsh column hybrid wavelet transform, they are also acceptable and give good robustness.

Table 20 shows MAE between embedded and extracted watermark against resizing attack when DCT-Walsh row wavelet and Walsh-DCT row wavelet generated from DCT and Walsh of different sizes are used to embed and extract the watermark.

Table 20 Comparison of MAE between embedded and extracted watermark against resizing attack using DCT-Walsh and Walsh-DCT hybrid wavelet row transform

Resize Type	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT	DCT-Walsh	Walsh-DCT
	64by4	**64by4**	**32by8**	**32by8**	**16by16**	**16by16**	**8by32**	**8by32**	**4by64**	**4by64**
Resize2	29.824	33.464	33.828	31.924	32.212	26.899	32.830	26.661	34.565	28.127
DFT-resize	1.058	1.445	1.355	1.693	1.314	1.903	1.468	1.791	1.624	2.252
grid resize	3.858	2.247	5.476	3.358	8.197	4.253	13.385	5.527	19.463	9.375

As can be seen from Table 20, for resizing using bicubic interpolation, frequent fluctuations are observed in performance of DCT-Walsh and Walsh-DCT row wavelet transforms when size of DCT and Walsh matrix is changed to obtain them. After an overall comparison of DCT-Walsh and Walsh-DCT wavelet transforms, Walsh-DCT can be concluded as more robust than DCT-Walsh row wavelet. For transform based resizing, except DFT other transforms when used for resizing give zero MAE. Resizing using DFT shows very small MAE for both DCT-Walsh and Walsh-DCT row wavelet transforms in which DCT-Walsh shows marginally better robustness. For resizing using grid based interpolation, DCT when used as local component transform with Walsh as global one, makes the proposed method more robust.

5.4.2. DCT-Haar and Haar-DCT hybrid wavelet transform

Table 21 and Table 22 show average mean absolute error against resizing attack when DCT-Haar and Haar-DCT wavelet transforms are used in column and versions respectively to insert watermark.

Table 21 Comparison of MAE between embedded and extracted watermark against resizing attack using DCT-Haar and Haar-DCT hybrid wavelet column transform

Resize Type	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT
	64by4	64by4	32by8	32by8	16by16	16by16	8by32	8by32	4by64	4by64
Resize2	28.532	32.038	28.705	29.494	27.912	27.028	29.962	28.178	27.994	28.875
DFT_resize2	1.230	1.755	1.129	1.798	1.086	1.566	1.128	1.536	1.091	1.793
grid resize2	4.319	2.968	3.048	3.365	3.029	4.398	3.087	5.305	3.179	8.566

For grid based resizing and for resizing using DFT, DCT when used as global and local component transform along with Haar, shows strong robustness. For other transforms like DCT, DST, Haar and Hartley transform DCT-Haar and Haar-DCT show excellent robustness with zero MAE. For resizing using bicubic interpolation, for different size combinations, DCT-Haar and Haar-DCT show continuous fluctuations.

Table 22 Comparison of MAE between embedded and extracted watermark against resizing attack using DCT-Haar and Haar-DCT hybrid wavelet row transform

Resize Type	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT	DCT-Haar	Haar-DCT
	64by4	64by4	32by8	32by8	16by16	16by16	8by32	8by32	4by64	4by64
Resize2	29.573	33.464	31.156	31.924	30.665	26.899	30.058	26.661	30.257	28.127
FFT_resize2	1.094	1.445	1.338	1.693	1.177	1.903	1.077	1.791	1.188	2.252
grid resize2	3.864	2.247	4.028	3.358	3.583	4.253	3.919	5.527	3.396	9.375

Observations for row version of DCT-Haar and Haar-DCT wavelet transforms against resizing attack are same as column version written above from Table 21.

5.4.3. DCT-DKT and DKT-DCT hybrid wavelet transform

Table 23 below shows summary of performance of DCT-DKT and DKT-DCT column wavelet transform against resizing attack. Similarly Table 24 summarizes performance of DCT-DKT and DKT-DCT row wavelet transform against resizing attack

Table 23 Comparison of MAE between embedded and extracted watermark against resizing attack using DCT-DKT and DKT-DCT hybrid wavelet column transform

Resize Type	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT
	64by4	64by4	32by8	32by8	16by16	16by16	8by32	8by32	4by64	4by64
Resize2	39.880	32.038	39.058	29.494	36.810	27.028	37.689	28.178	38.973	28.875
DFTresize2	2.327	1.755	1.905	1.798	1.708	1.566	1.547	1.536	1.622	1.793
gridresize2	3.061	2.968	2.066	3.365	2.045	4.398	1.901	5.305	2.119	8.566

Table 24 Comparison of MAE between embedded and extracted watermark against resizing attack using DCT-DKT and DKT-DCT hybrid wavelet row transform

Resize Type	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT	DCT-DKT	DKT-DCT
	64by4	64by4	32by8	32by8	16by16	16by16	8by32	8by32	4by64	4by64
Resize2	39.547	33.464	37.810	31.924	39.627	26.899	38.608	26.661	39.781	28.127
DFT_resize2	2.119	1.445	1.828	1.693	1.795	1.903	1.581	1.791	1.735	2.252
grid resize2	2.764	2.247	2.479	3.358	2.097	4.253	1.997	5.527	2.479	9.375

From Table 23 and Table 24 it can be observed that row as well as column versions of DCT-DKT and DKT-DCT, strong robustness is observed against resizing using DFT and resizing using grid based interpolation. For other transforms used for resizing, both column and row versions show excellent robustness with zero MAE. For bicubic interpolation based resizing, DCT when used as local component transform with DKT as global, proves to be better than using DCT-DKT wavelet transform.

6. CONCLUSIONS

Sinusoidal transform DCT and non-sinusoidal transforms Walsh, Haar and DKT are used to generate hybrid wavelet transform. DCT is combined with one of the remaining non-sinusoidal transforms to generate hybrid wavelet transform. Different sizes of component transform are required to sustain against different types of attacks in the proposed method. Proposed method is found to be highly robust against cropping, resizing using transforms, resizing using grid based interpolation and noise addition attacks when DCT is used as local component transform. Using DCT as global component is proved robust against compression, resizing using transforms, resizing using grid based interpolation and noise addition attacks.

REFERENCES

[1] Wai Chu, (2003) "DCT-Based Image Watermarking Using Subsampling", IEEE transactions on multimedia, vol. 5, no. 1, pp. 34-38.

[2] Adrian G. Bor_s and Ioannis Pitas, (1998) "Image watermarking using block site selection and DCT domain constraints", Optics Express, Vol. 3, No. 12, pp.512-523.

[3] Rajesh Kannan Megalingam, Mithun Muralidharan Nair, Rahul Srikumar, Venkat Krishnan Balasubramanian and Vineeth Sarma Venugopala Sarma, (2010) "A Comparative Study on Performance of Novel, Robust Spatial Domain Digital Image Watermarking with DCT Based Watermarking", International Journal of Computer Theory and Engineering, Vol. 2, No. 4, pp. 647-653.

[4] Dr. B. Eswara Reddy, P. Harini, S. Maruthu Perumal & Dr. V. Vijaya Kumar, (2011) "A New Wavelet Based Digital Watermarking Method for Authenticated Mobile Signals", International Journal of Image Processing (IJIP), Volume (5): Issue (1), pp. 13-24.

[5] Nagaraj V. Dharwadkar & B. B. Amberker, "Determining the Efficient Sub band Coefficients of Biorthogonal Wavelet for Grey level Image Watermarking", International Journal of Image Processing Volume (4): Issue (2), pp. 89-105.

[6] Yiwei Wang, John F. Doherty, and Robert E. Van Dyck, (2002) "A Wavelet-Based Watermarking Algorithm for Ownership Verification of Digital Images", IEEE transactions on image processing, vol. 11, no. 2, pp.77-88.

[7] Ruizhen Liu and Tieniu Tan, (2002) "An SVD-Based Watermarking Scheme for Protecting Rightful Ownership", IEEE transactions on multimedia, vol. 4, no.1, pp. 121-128.

[8] Kapre Bhagyashri, Joshi, M.Y., (2010) "Robust image watermarking based on singular value decomposition and discrete wavelet transform", Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on, vol.5, no., pp.337-341.

[9] G. Bhatnagar, B. Raman, Q. M. J. Wu, (2012) " Robust watermarking using fractional wavelet packet transform", IET Image Processing, vol. 6, issue 4, pp. 386-397.

[10] Ahmed Abdulfetah, Xingming Sun, Hengfu Yang and Nur Mohammad, (2010) "Robust adaptive image watermarking using visual models in DWT and DCT domain", Information Technology journal 9(3), pp. 460-466.

[11] Mohsen Kariman Khorasani, Mohammad Mojtaba Sheikholeslami, (2012) "An DWT-SVD Based Digital Image Watermarking Using a Novel Wavelet Analysis Function", Fourth International Conference on Computational Intelligence, Communication Systems and Networks, pp. 254-256.

[12] A. Sverdlov, S. Dexter and A. M. Eskicioglu, (2004) "Robust DCT-SVD domain image watermarking for copyright protection: Embedding data in all frequencies", in proc. of Multimedia and Security Workshop, ACM press, pp. 166-174.

[13] Navas K.A., Ajay, M.C. Lakshmi, M., Archana T.S., Sasikumar M., (2008) "DWT-DCT-SVD based watermarking", Third International Conference on Communication Systems Software and Middleware and Workshops, pp.271-274.
[14] A. Umaamaheshvari and K. Thanushkodi, (2013) "A robust digital watermarking technique based on feature and transform method", scientific research and essays, vol. 8(32), pp. 1584-1593.
[15] Haijun luo1, Xingming sun, Hengfu yang, Zhihua Xia, (2011) "A robust image watermarking based on image restoration using SIFT, radio engineering", Vol. 20, No. 2, pp. 525-532.
[16] Chih-Chin L, Cheng-Chih T, (2010) "Digital Image Watermarking Using Discrete Wavelet Transform and Singular Value Decomposition", IEEE Transaction on Instrumentation Measurement. 59:3060-3063.
[17] Lagzian S, Soryani M, Fathy M (2011). A new robust watermarking scheme based on RDWT-SVD. Int. J. Int. Inf. Processing (2):22-29
[18] Li Y, (2009) "A Water Marking Algorithm of Digital Image Based on DWT-SVD" 1st International Conference on Information Science and Engineering (ICISE), pp. 5307-5310.
[19] Chang CC, Tsai P, Lin CC (2005) "SVD-based digital image watermarking scheme", Patt. Recog. Lett. 26:1577-1586.
[20] Rastegar S, Namazi F, Yaghmaie K, Aliabadian A, (2011) " Hybrid watermarking algorithm based on Singular Value Decomposition and Radon transform", AEU – International Journal of Electronic Communication, 65:658-663.
[21] Qingtang Su, Xianxi Liu, Wenchao Yang, (2009) "A Watermarking Algorithm for Color Image Based on YIQ Color Space and Integer Wavelet Transform", In IEEE International Conference on Image Analysis and Signal Processing, pp. 70-73.
[22] Ying Zhang, Jiqin Wang, Xuebo Chen, "Watermarking technique based on wavelet transform for color images" 24th Chinese Control and Decision Conference (CCDC), 2012, pp. 1909-1913.
[23] H. B. Kekre, Tanuja Sarode, Sudeep Thepade. "Inception of Hybrid Wavelet Transform using Two Orthogonal Transforms and It's use for Image Compression", International Journal of Computer Science and Information Security, volume 9, No. 6, pp. 80-87, 2011.
[24] H. B. Kekre and Tanuja K. Sarode. "Fast Codebook Generation Algorithm for Color Images using Vector Quantization." International Journal of Computer Science and Information Technology 1.1 (2009): 7-12.
[25] H. B. Kekre, Tanuja Sarode, Sudeep Thepade, "Grid based image scaling technique", International Journal of Computer Science and Applications, Volume 1, No. 2, pp. 95-98, August 2008
[26] Dr. H. B. Kekre, Dr. Tanuja Sarode, Shachi Natu, "Image Zooming using Sinusoidal Transforms like Hartley, DFT, DCT, DST and Real Fourier Transform", International journal of computer science and information security Vol. 12 No. 7, pp. 11-16, July 2014.