COMPARATIVE ANALYSIS OF USE OF CURVELET AND WAVELET FOR ROBUST INVISIBLE WATERMARKING

Harshula Tulapurkar[1], B. Krishna Mohan[2] , Varsha Turkar[3]

[1] CSRE, Indian Institute of Technology-Bombay, Powai, Mumbai-400 076, India
Email: harshulat@yahoo.com
[2]CSRE, Indian Institute of Technology-Bombay, Powai, Mumbai-400 076, India,
Email: bkmohan@csre.iitb.ac.in
[3]Dept. of Information Technology, Vidyalankar Institute of Technology, Wadala. Mumbai India,
Email: varshaturkar@gmail.com

KEY WORDS:- Curvelet, Wavelet, Texture, Watermark, Satellite, Haralick, SURF features, Robust

ABSTRACT - In this paper, a robust invisible watermarking algorithm for satellite imagery using Curvelet Transform is proposed. Haralick Co-occurrence texture features (Haralick. et. al., 1973) are used to identify the area for watermarking in the host image. Host image is tiled into smaller non-overlapping blocks. Based on the Haralick texture feature chosen, blocks with high value of chosen texture feature were selected for embedding. Thus, multiple watermarks are embedded in any given image. There are some unstable Curvelet coefficients so a little change of the image will result in a big change of these coefficients. These unstable factors can influence the extracting of watermark. Hence selection of position of embedding in the transformed domain plays a very important role in robustness of the embedding process. This algorithm encourages use of edges and curves for embedding watermarks. The experimental results show that watermark using proposed algorithm is robust against common attacks like Brightness, Contrast, Saturation, Tint adjustments, Low pass Filtering, JPEG Compression attack, Gaussian Noise attack and Laplacian Filtering, Symmetrical and Asymmetrical Image Cropping, Geometric Attacks like scaling and Rotation. Use of SURF features (Chincha. et. al., 2011) too ensures robustness to geometric attacks. Comparative study with wavelet watermarking algorithm proposed by Bazargani (Bazargani. et al., 2012) shows the improvements in results obtained.

I. INTRODUCTION

Geospatial data or geographic data identifies the geographic location of features and boundaries on Earth, such as natural or constructed features, oceans, and more. Geographic Information has long been used in diverse applications for commercial, scientific and defence use. Satellite imagery forms one of the most frequently used geographic data and with the level of detail that this imagery can reveal in today's technology, the data has significant strategic and commercial value. Therefore, there is a compelling need for ensuring of authenticity and protection of ownership. For providing security of digital data various techniques are used like encryption, decryption, cryptography, and digital watermarking. Digital watermarking is a technique of embedding selected user information into the digital content like image, video, speech, music etc. The watermarking algorithm should be able to detect intentional tampering of the original data and retain its integrity within the content even after various manipulation attacks like compression, enhancement, cropping, filtering etc.

Most natural images/signals exhibit line-like edges, and discontinuities across curves called curve singularities. Candes and Donoho (Candes. et. al., 2004) (Candes. et. al., 2006) proposed a multi-resolution geometric analysis (MGA), named Curvelet transform that not only considers a multi-scale time (or space)-frequency local partition but also makes use of the direction of features. The Curvelet transform directly takes edges as the basic representation element; it provides optimally sparse representations of objects along edges. Candes and Donoho suggested two strategies, namely Unequally-Spaced Fast Fourier Transform (USFFT) and Frequency wrapping. The Wrapping based Curvelet transform technique is conceptually simpler, faster and less redundant than the previous techniques.

A preliminary literature review of past studies shows that watermarking techniques in frequency domain are primarily focused on transforms like Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT). Lot of work has been done in field of watermarking in Wavelet Domain. Wavelet based watermarking techniques are robust, popular and efficient. However little work has been made on invisible watermarking using Discrete Curvelet Transform. In this paper, a robust invisible watermarking technique is proposed using the wrapping FDCT method and a comparative study with wavelet transform is presented. Primary Objective of watermarking technique proposed in this paper is copyright protection and image authentication.

II. CURVELET TRANSFORM

Discrete Fourier Transform (DFT) is the most common and powerful procedure to analyse, manipulate and synthesize digital signals. Though the Fourier expansion provides frequency resolution but it does not provide time resolution. The wavelet transforms or wavelet analysis overcomes this shortcoming of the Fourier Transform by giving a time-frequency joint representation. The idea behind these time-frequency joint representations is to cut the signal of interest into several parts and then separately analysing each part. This gives more information about the when and where of different frequency components.
Natural images usually have line-like edges, i.e., discontinuities across curves, which are called line or curve singularities. However, wavelets cannot represent line singularities and so Curvelet Transform is used. The Curvelet transform is a multi-scale transform like the wavelet transform, with frame elements indexed by scale and location parameters. It preserves the same time frequency localization property as for wavelets and at the same time Curvelet become directional. It acts like a band-pass filter. In addition, anisotropic scaling principle, which is quite different from the isotropic scaling of wavelets, helps in sparse representation. The elements obey a special parabolic scaling law, defined by width ≈ length^2. So instead of square representation it is now rectangular representation (**Figure. 1**). By changing the scale location and orientation the multi-scalar coefficients can be obtained as shown in **Figure 2** to **4**.

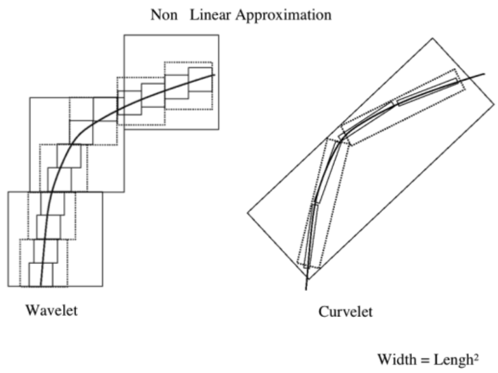


Figure 1 Parabolic scaling, non-linear approximation (Starck et.al., 2010)

In Second generation transform Curvelet DCTG2, first 2D Fast Fourier Transform (FFT) of the image is taken. The 2D Fourier frequency plane is then divided into wedges. The partitioning of the Fourier plane into dyadic squares and angular divisions results in parabolic shaped wedges. Each square represents a scale and acts like a bandpass filter and the angular divisions partition the band passed image into different angles or orientations.

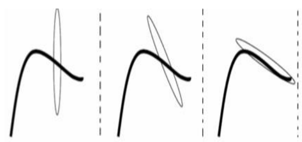


Figure 2 Curvelet with fixed scale, fixed orientation but variable location (Starck et.al., 2010)

Figure 3 Curvelet with fixed scale, fixed location but variable orientation (Starck et.al., 2010)

Figure 4 Curvelet with fixed orientation, fixed location but variable scale (Starck et.al., 2010)

Curvelab 2.1.3 software package implements the FDCT_WRAPPING algorithm. The DCTG2 implementation can assign either wavelets or Curvelets at the finest scale. Scale (resolution) and angles (no of orientation) can be defined to get varying level of resolution. For the algorithm proposed in this paper, 4 scales - 8 angles wrapping Curvelet transform with Curvelets at the finest level is used. The output of a 4 scale - 8 angles Wrapping Curvelet transform results in

Curvelet Transform output = {1x1} {1x8} {1x16} {1x16}

{1x1} – Represents the low frequency components
{1x8} - Scale 2. Each of the 8 columns represents data for that angle (wedge)
{1X16} - Scale 3. Each of the 16 columns represents data for that angle (wedge)
{1x16}- Scale 4. Each of the 16 columns represents data for that angle (wedge)

III. EMBEDDING ALGORITHM

This algorithm incorporates semi-blind robust watermark extraction wherein the original image is not available for extraction; hence a key containing detail about embedding is attached to the watermarked image. Watermark is embedded in the third band of the image.

1. Resize the Host Image to MXN such that M and N are power of 2.
2. Choose a watermark image (A x B) and convert it into binary. Serialize the watermark W_s. Ws = {Wk = 1, 2, 3, …, A × B; Wk {−1,1}}
3. Split the Host image into smaller non-overlapping blocks (e.g image size 2048 x 2048 block size is 512 x 512; image size 512 x 512 block size is 64 x 64)
4. Extract co-occurrence texture features for each block. Mark blocks that have high texture feature (as specified by user) for embedding. Deselect 1 row and 1 column of blocks from all the four edges to ensure that no watermarks are lost when borders are cropped.
5. For each marked block do the following
   a. Apply Wrapping Curvelet Transform (fdct_wrapping) with finest level as wavelet, 4 scales and 8 angles. C denotes this Curvelet Transform.
   b. Implement LOCEDGES logic that selects locations and orientation for embedding watermarks in Scale 3 Curvelet coefficients (section III A). arr_max contains orientation and location for embedding $W_k = 1$ and arr_min contains orientation and location for $W_k = -1$.
   c. For each location in level 3, S_{3O} (i, j), O indicates orientation; there are 4 dependent locations in Scale 4, $S_{4O.}$ (2i, 2j; 2i, 2j-1; 2i-1, 2j; 2i-1, 2j-1). These are called child nodes. Find Max (max_node) and Min (min_node) of these 4 child nodes.
   d. **If** $W_k = 1$
      select location (i, j) and orientation (O) from arr_max
      C {1,3} {1, O} (i, j) = max_node + alpha
      else
      select location (i,j) and orientation (O) from arr_max
      C {1,3} {1, O} (i,j)= min_node - alpha
      End if
      The chosen value of alpha = 160, which can be adjusted to change the strength of invisibility.
   e. Apply Inverse wrapping Curvelet transform to get watermarked image.

6. Concatenate all the split blocks (modified and unmodified) to form Host image with multiple Generate a key by using RSA encryption to encrypt following data
   a. SURF features of the watermarked image. This includes features and valid points for each descriptor. This is useful for visualizing the descriptor orientation.
   b. Watermarked location, orientation and original coefficient value of each watermarked block.
   c. Blocks that are watermarked.
   d. Original size of the Host Image and watermark image.

A. LOCEDGES

Logic for selecting orientation and locations for embedding watermarks This algorithm chooses edges (high value coefficients) for embedding 1's in watermark and low value coefficients for embedding -1's in watermark.

1. Scan the entire Curvelet coefficients across all orientations in Level 3 and create an array arr_level_max that holds value of maximum coefficient in each orientation.
2. Select the orientation from arr_level_max with maximum coefficient value max_value.
3. Set threshold T_{max}=0.5*max_value. Create an array arr_max that holds orientation and location of all coefficient values greater than $T_{max.}$
4. If no. of locations found is less than total no of 1s in watermark repeat step 3 with next highest value from arr_level_max
5. For identifying locations to embed -1s set T_{min}=0 and choose all locations in a given orientation which are less than $T_{min.}$

IV. EXTRACTION ALGORITHM

1. Check for Geometric Attack (section IV A).
2. Split the Embedded Host Image into smaller non-overlapping blocks with block size same as that in the embedding algorithm.
3. Extract the watermarked blocks from the key.
4. Apply Wrapping Curvelet Transform (fdct_wrapping) with finest level as wavelet, 4 scales and 8 angles. $C_{extract}$ denotes this Curvelet Transform.
5. For every watermarked block

a. Extract the Watermarked location (i, j), orientation (O) and original value (org_val) of each watermarked block.
b. If $C_{extract}$. {1,3} {1,0} (i,j) > org_val, W_k=1
Else W_k=-1.

6. Reshape the serial Watermark into 2D depending on size extracted from key.

A. ALGORITHM TO CHECK GEOMETRIC ATTACK

1. Extract the SURF Features of the original embedded image from the key.
2. Find the SURF Features of the altered embedded image.
3. Find the matching points between the original and altered images using estimate Geometric Transform.
4. Restore the image using imwarp. (This takes care of the size and rotation attacks)

Another technique for detecting Geometric attacks is using Radon Transform where the maximum value in the Radon transform can be used to detect scale change and the rotation can be detected by location of the maximum value. However, this cannot detect any asymmetrical cropping of the image and hence algorithm using SURF features provides a robust solution to detecting geometric attacks and asymmetrical cropping. SIFT features could also be used, however this was not explored.

V. RESULT AND DISCUSSION

A satellite image SAT5 (1856x1404) (SIP LAB, CSRE, IIT Bombay) was used as host image and watermark image was of size 7x22. Robustness of algorithm for a variety of watermarking attacks was tested. Same Embedding Logic was incorporated in Wavelet transform as well as Curvelet transform and then the performance was compared.

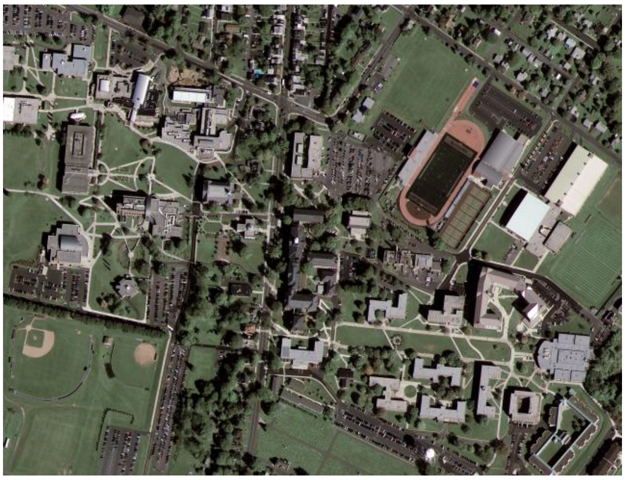


Figure 5 Host Image SAT5 (1856 x1404)

Figure 6 Watermark Image (7x22)

Haralick texture feature ‘Contrast’ was used to identify watermarking area. Response to 6 attacks viz. Compression, Gaussian Noise, Geometric attack, Contrast adjustment, smoothing (neighbourhood filtering), is tabulated below (**Table II to VII**). Table VIII shows samples of retrieved watermarks in various attacks.

Notations	***Details***
#W embedded	No of watermarks embedded
#W retrieved	No of watermarks extracted correctly
CPSNR	Cumulative PSNR (For all watermarks)
CMSE	Cumulative MSE (For all watermarks)

Table I Notations used in tables II to VI

% Noise in dB	***Curvelet Watermarking***				***Wavelet Watermarking***			
	#W embedded	***#W extracted***	***CPSNR***	***CMSE***	***#W embedded***	***#W extracted***	***CPSNR***	***CMSE***
10	8	0	65.36	4.7	11	0	61.74	11.45
15	8	8	Infinity	0	11	0	63.13	10.12

20	8	8	Infinity	0	11	0	64.14	10.38
25	8	8	Infinity	0	11	0	64.34	5.3
30	8	8	Infinity	0	11	0	65.36	4.4

Table II Comparative analysis of Gaussian attack- Curvelet vs Wavelet watermarking

Scaling (S), Rotation (R) Factor		***Curvelet Watermarking***				***Wavelet Watermarking***			
		#W embedded	***#W extracted***	***CPSNR***	***CMSE***	***#W embedded***	***#W extracted***	***CPSNR***	***CMSE***
S	***R***								
1	2	8	7	70.06	1.20	11	1	63.07	5.07
1	4	8	6	67.62	3.89	11	0	63.12	6.49
1	6	8	4	63.11	4.20	11	2	63.14	5.54
1	10	8	4	64.17	4.80	11	0	64.62	6.84
1.2	2	8	6	60.45	1.56	11	0	62.69	7.32
1.2	4	8	6	58.46	4.10	11	0	61.50	4.40
1.2	6	8	4	56.12	4.60	11	0	63.34	5.19
0.21	10	8	4	55.43	5.10	11	0	63.98	6.96

Table III Comparative analysis of Geometric attack- Curvelet vs Wavelet watermarking

Technique	***Curvelet Watermarking***				***Wavelet Watermarking***			
	#W embedded	***#W extracted***	***CPSNR***	***CMSE***	***#W embedded***	***#W extracted***	***CPSNR***	***CMSE***
Histogram equalization	8	5	70.06	1.6	11	0	55.12	30.46
Adjusting Image Intensity to increase contrast	8	8	Infinity	0	11	0	60.06	8.50
Contrast-limited adaptive histogram equalization	8	6	66.96	0.81	11	0	58.29	22.43

Table IV Comparative analysis of Contrast Enhancement attack- Curvelet vs Wavelet watermarking

Filter	***Curvelet Watermarking***				***Wavelet Watermarking***			
	#W embedded	***#W extracted***	***CPSNR***	***CMSE***	***#W embedded***	***#W extracted***	***CPSNR***	***CMSE***
Averaging	8	0	57.14	101	11	0	68.32	1.89
Gaussian LPF (f=10)	8	0	Infinity	0	11	1	70.06	1.89
Gaussian LPF (f=100)	8	0	57.18	112	11	1	70.06	1.89
Gaussian LPF (f=1k)	8	0	57.19	115	11	1	70.06	1.89
Gaussian LPF (f=10k)	8	0	57.19	116	11	1	70.06	1.89

Table V Comparative analysis of Low Pass filtering attack- Curvelet vs Wavelet watermarking

Cropping	***Curvelet Watermarking***				***Wavelet Watermarking***			
	#W embedded	***#W extracted***	***CPSNR***	***CMSE***	***#W embedded***	***#W extracted***	***CPSNR***	***CMSE***
10% Left	8	8	Infinity	0	11	11	Infinity	0
10% Right	8	8	Infinity	0	11	11	Infinity	0
10% Top	8	8	Infinity	0	11	11	Infinity	0
10% Bottom	8	8	Infinity	0	11	11	Infinity	0

Table VI Comparative analysis of Cropping attack- Curvelet vs Wavelet watermarking

Filter	*#W embedded*	*#W extracted*	*CPSNR*	*CMSE*
Averaging	1	0	61.55	9.09
Gaussian LPF (f=10)	1	0	62.22	7.79
Gaussian LPF (f=100)	1	0	70.06	2.28
Gaussian LPF (f=1k)	1	0	60.90	10.3
Gaussian LPF (f=10k)	1	0	62.22	7.79

Table VII Results for various attacking for Hybrid Curvelet Watermarking Algorithm

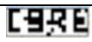
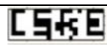
Attack	*Curvelet Watermarking*				*Wavelet Watermarking*			
	#W embedded	*#W extracted*	*Max MSE*	*Watermark with Max Error*	*#W embedded*	*#W extracted*	*Max MSE*	*Watermark with Max Error*
20% Compression	8	0	7.79		11	0	44.15	
10 dB Gaussian Noise	8	0	3.90		11	0	11.45	
Average filter*	1	0	9.09		11	0	1.89	
Contrast Adjustment Histogram Equalization	8	5	1.60		11	0	48.12	
Scaling and Rotation (S=1 & R=2)	8	7	1.20		11	1	9.09	

Table VIII Samples of retrieved watermarks in various attacks

From the above observation, it is evident that the Curvelet watermarking algorithm fails the low pass filtering and averaging attacks as edges are used for embedding the watermarking. Filtering causes the edges to be is smoothened and so watermarks are lost. Therefore, a hybrid-watermarking algorithm is introduced wherein 80% of the watermarks are embedded using the above logic whereas for the 20% watermarks locations are chosen where there is not much variation thus they can with stand averaging and Low pass filtering attacks. Embedding in high frequency coefficients offers better imperceptibility, while low frequency coefficients provide high robustness against Filtering attacks. From **Figure. 7** and **Figure. 8** below it is seen that watermarking does not cause any visible change in the image.

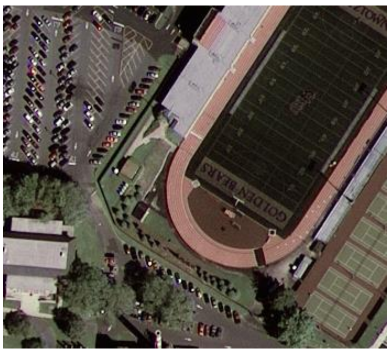


Figure 7 – Host Image without watermark

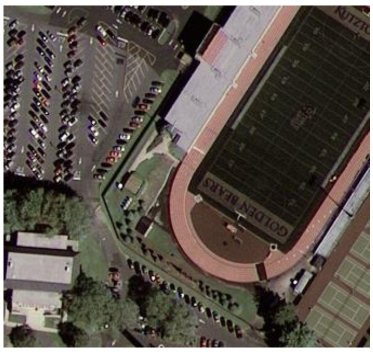


Figure 8 Watermarked Image

The difference between host image and watermark image is shown below in Figure 9 and Figure 10.

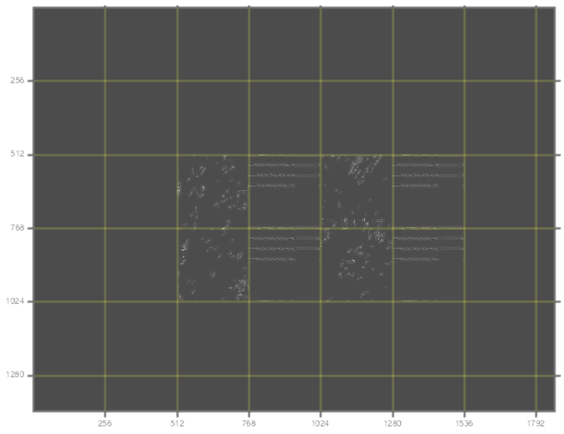


Figure 9 Difference between the Host image and Curvelet Watermarked image. Note that there are 8 watermarked blocks and the dashes indicate watermarking

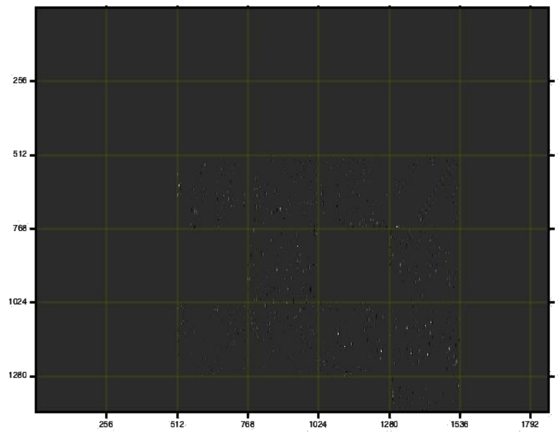


Figure 10 Difference between the Host image and Wavelet Watermarked image. (11 watermarked blocks and the dashes indicate watermarking)

V. CONCLUSION

This new approach for embedding Invisible Watermarking using Curvelet transform shows improved performance over wavelet transform when embedding logic is same. Robustness against variety of attacks is due to use of texture features to select blocks combined with selection of appropriate locations for embedding. Use of SURF features serves in synchronizing embedding location, which helps in detecting and recovering from geometric attacks resulting in negligible MSE. The proposed technique can also be used to watermarked multiband images.

It has been observed that the algorithm is not capable of handling Low pass filtering and average filtering attack. This is as expected because edges in the images are used for embedding watermark. To overcome this drawback a hybrid embedding logic is incorporated where 20% watermarks are embedded in locations that do not have sharp edges.80% watermarks are embedded in edges, this ensures robustness against filtering attacks.

.

REFERENCES

Bazargani, Ebrahimi and Dianat, 2012. “Digital Image Watermarking in Wavelet, Contourlet and Curvelet Domains” *Journal of Basic and Applied Scientific Research,* ISSN 2090-4304, pp. 11296-11308.

Candes E.J and Donoho D.J, 2004. “New tight frames of Curvelets and optimal representations of objects with piecewise-C^2 singularities”. *Communications on Pure and Applied Mathematics*, vol. 57, 219-266.

Candes E.J, Demanet L., Donoho D.L, Ying L, 2006. “Fast Discrete Curvelet Transforms”, Multiscale Modeling & Simulation, 5(3), pp. 861-899.

Chincha R and Tian Y, 2011,"Finding objects for blind people based on SURF features," *IEEE International Conference on Bioinformatics and Biomedicine Workshops (BIBMW)*, Atlanta, GA, pp. 526-527.

Haralick R. M, Shanmugam K. and Dinstein I., Nov. 1973. "Textural Features for Image Classification," in *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-3, no. 6, pp. 610-621.

Starck Jean-Luc, Murtagh Fionn, Fadili Jalal M., 2010.“The Ridgelet and Curvelet Transforms”, in *Sparse Image and Ridgelet Processing: Wavelets, Curvelets, Morphological Diversity,* 1st ed., Cambridge University Press, NY, USA, pp. 89-118.