Advanced Digital Image Compression Technique Using Curvelet Transform

D. Vishnuvardhan[1], Sreenivasan. B[2], I. Suneetha[3]
[1]M. Tech (DSCE) Student, ECE Department, AITS, Tirupati
[2]Assistant Professor ECE Department, AITS, Tirupati
[3]Associate Professor, ECE Department, AITS, Tirupat

Abstract
In real time applications Image Compression plays a major role in storage memory and transmission bandwidth. As many compression methods are existing, selection of compression methods depends upon the quality in image reconstruction. Two popular compression methods are Wavelet and Ridgelet methods. Wavelet method has affected by the blocking artifacts and Ridgelet is designed to handle only line and edge singularities. This paper proposes a compression method using curvelet transform. The Curvelet Transform overcomes the drawbacks of above two methods and also has improvement in PSNR because of adaptive threshold.

Key Words: PSNR, Retained Coefficients, Ridgelet Transform

I. INTRODUCTION

Data compression[1] is the process of converting a data stream into another data stream that has smaller size. Different compression techniques have been developed for the past two decades. The basic principle of compression is to remove the redundancy in the source data. Compression basically is of two types – lossless and lossy[2]. This paper is connected with lossy image compression of color images. Many techniques exist for the compression of color images. A Block Partitioning Coder algorithm based on modified SPHIT Method based on Vector Quantization (VQ) has been an efficient method for image compression. The paper describes a VQ based, Fast Clustering Algorithm[3].Color image compression using the CT is not yet explored. This work explores use of Curvelet transform[10] and comparing it with the wavelet transform and Ridgelet transform[7] for color image compression

II. CURVELET TRANSFORM (CT):

The work throughout in two dimensions, i.e.,R^2, with spatial variable x, with ω a frequency domain variable, and with r and θ polar coordinates in the frequency-domain. Start with a pair of windows W(r) and V (t), which we will call the "radial window" and "angular window," respectively. These are smooth, nonnegative and real-valued, with W taking positive real arguments and supported on $r \in (\frac{1}{2}, 2)$ and V taking real arguments and supported ont $\in [-1, 1]$.These windows will always obey the admissibility conditions:

$$\sum_{j=\infty}^{\infty} W^2(2^j r) = 1, \quad r \in \left(\frac{3}{4}, \frac{3}{2}\right);$$
$$\sum_{j=\infty}^{\infty} V^2(t-l) = 1, \quad t \in \left(-\frac{1}{2}, \frac{1}{2}\right) \ldots \quad (1)$$

Now, for each $j \geq j_0$, introducing the frequency window U_j defined in the Fourier domain by

$$U_j(r,\theta) = 2^{-\frac{3j}{4}} W(2^{-j} r) V\left(\frac{2^{\lfloor j/2 \rfloor}\theta}{2\pi}\right) \ldots . \quad (2)$$

where$\lfloor j/2 \rfloor$ is the integer part of j/2. Thus the support of U_j is a polar "wedge" defined by the support of W and V, the radial and angular windows, applied with scale-dependent window widths in each direction. To obtain real-valued curvelets, work with the symmetrized version of (2.3), namely,

$$U_j(r,\theta) + U_j(r,\theta+\pi).$$

Define the waveform $\varphi_j(x)$ by means of its Fourier transform$\hat{\varphi}_j(\omega) = U_j(\omega)$. $\varphi_j$is a "mother" curvelet in the sense that all curvelets at scale 2^{-j} are obtained by rotations and translations of φ_j. Introduce

The equispaced sequence of rotation angles $\theta_l = 2\pi . 2^{-\lfloor j/2 \rfloor} . l$, with $l = 0,1, \ldots$such that$0 \leq \theta_l < 2\pi$ (note that the spacing between consecutive angles is scale-dependent),and the sequence of translation parameters $k = (k_1, k_2) \in Z^2$.With these notations, defining curvelets (as function of $x = (x_1, x_2)$) at scale 2^{-j}, orientation θ_l and position $x_k^{(j,l)} = R_{\theta_l}^{-1}\left(k_1 . 2^{-j}, k_2 . 2^{-\frac{j}{2}}\right)$ by

$$\varphi_{j,l,k}(x) = \varphi_j\left(R_{\theta_l}(x - x_k^{(j,l)})\right) \quad \ldots .(3)$$

whereR_θ is the rotation by θ radians and R_θ^{-1} its inverse (also its transpose).

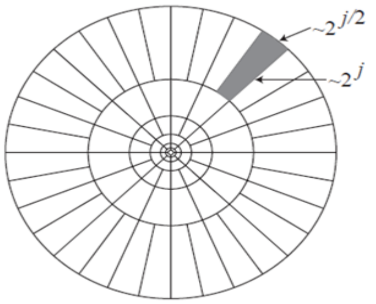

Fig. 1:Curvelet tiling of space and frequency.

Then the ridgelet transform is the application of a 1-D wavelet transform to the slices (also referred to as projections) of the Radon transform, and is denoted as

$CRT_f(a, b, \theta) = \int_R \psi_{a,b}(t) R_f(\theta, t) dt$ .... (4)

By definition, the Finite Ridgelet Transform (FRIT) canbe indicated as

$FRIT_f\ [k, m] = FRAT_f\,[k,.], w_m^{(k)}\,[.]$ ... (5)

To overcome the weakness of wavelets in higher dimensions, ridgelets can be used effectively to deal with line singularities in 2-D. The idea is to map a line singularity into a point singularity[5] using the Radon transform. Then, the wavelet transform can be used to effectively handle the point singularity. In short, the Ridgelet transform is the application of 1 -D wavelet transform to the slices of the radon transform.

The Ridgelet transform is used for many applications mainly for the edge representation in natural images, in motion compensated images. Comparison of JPEG, Wavelet and Ridgelet transform for compression is implemented. It is possible to convert from RGB to YCbCr and vice versa. In this paper, we use of **YCbCr** format.

III. ANALYSIS

The analysis presented in this paper is based on three transforms namely Wavelet, Ridgelet and Curvelet Transforms. The color images are selected from the set of standard images. Results for Compression metrics, the Root Mean Square Error (RMSE) and PSNR are tabulated.

A specific thresholding method used in each of Ridgelet and Wavelet case is described as follows:

i) For Wavelet Transform Algorithm:
Step 1: Convert input image from RGB to YCbCr Format.

Step 2: Separate the components Y, Cb, Cr.

Step 3: Process Y component.
3.1: Decompose Y component using discrete wavelet
3.2: Threshold subbands –*a* and *h*, *v*, *d* at all levels.
For a subbands(i, j) , the threshold value,
(i) m for a3,
(ii)2*m for h3, v3, d3
(iii)3*m for h2, v2, d2 and
(iv) 4*m for h1, v1, d1.

Step 4: Apply inverse wavelet transform to thresholded coefficient matrix - $\boldsymbol{w_{th}}$ to get the reconstructed image component Y.

Step 5: Repeat steps 3 and 4 for Cb and Cr components.

Step 6: Calculate MSE and PSNR between original and reconstructed image.
The calculations for the measures are defined as follows:
1. Mean Square Error (MSE) is a distortion measure for lossy compression. Mean square Error between two images is given as,
MSE=$\frac{1}{K}\sum_{i=1}^{k}(P_i - Q_i)^2$,Root mean square
RMSE = $\sqrt{MSE}$Where, P_i-Original Image
Q_i – Reconstructed image data, k size of image

2. Peak Signal to Noise Ratio (PSNR) is defined as,

$$\mathbf{PSNR} = 20\log_{10}\left(\frac{max_i|P_i|}{RMSE}\right)$$

3. Percentage of retained coefficients (% Of RC) is defined as
% RC = no. of non-zero coefficients/ total no. of coefficients×100.

ii) For Ridgelet Transform Algorithm:
Step 1: Transform input image from RGB to YCbCr format.

Step 2: Separate the components Y, Cb, Cr.

Step 3: Process the Y component.
3.1: Decompose Y component using ridgelet.
3.2: The multiplier values used for
The subbands-
(i) m for the most significant band 1(at level 3)
(ii) 2*m for band 2
(iii)3*m for band 3
(iv) 4*m for band 4
3.3: For Ridgelet Transformed coefficient matrix of a subband is $r(i, j)$, threshold value for column j,

$$\mathbf{v(j) = multiplier}*\left(\frac{1}{row}\right)\sum_{i=1}^{row}|r(i,j)|$$

Step 4:Invert the image from thresholded coefficient matrix - $\mathbf{r}_{th}$ to get reconstructed Y component.

Step 5: Repeat steps 3 and 4 for Cb and Cr components.

Step 6: Calculate MSE and PSNR between original and reconstructed image.

iii) For Curvelet Transform Algorithm*:*
Step 1: Transform input image from RGB to YCbCr format.

Step 2: Separate the components Y, Cb, Cr.

Step 3: Process the Y component.
3.1: Decompose Y component using Curvelet.
3.2: Threshold the different subbands, s(i,j)by value
$v = \text{multiplier}*(1/n)\Sigma_{i,j} |s(i,j)|, n = i*j$.
3.3 Retain the curvelet coefficients above the threshold value ***v*** *to form* **C*th.***

Step 4: Invert the image from thresholded coefficient matrix – **C**th to get reconstructed Y component.

Step 5: Repeat steps 3 and 4 for Cb and Cr components.

Step 6: Calculate MSE and PSNR between original and reconstructed image.

IV. RESULTS

The algorithms explained in section III have been applied to the standard images. Compression results with values of PSNR (Peak Signal to Nose Ratio) are given in Fig. 2, 3 ,4 and Table .1. Result images are given in Fig. 4. Fig.1,2,3 shows the graph of % of RC versus PSNR for three images, F1, F2 and F3 respectively. From Fig. 1,2 and 3 it is clear that the PSNR for wavelet based compression is better only for the higher % of RC, in the range 90 – 70 (lower compression ratio).When we decrease the % of RC, which indicates higher compression, the Curvelet transform shows higher PSNR values compared to ridgelet transform. It is seen that the PSNR using the Curvelet Transform is better than the Wavelet and Ridgelet case over a wider range of % of RC. Even with special thresholding method, the higher PSNR values can be obtained with Curvelet transform. . Wavelet transform gives better PSNR than Ridgelet only for the 90% and 80% of RC. After that its PSNR performance suddenly drops below Ridgelet, and Curvelet.

Comparison of PSNR among transforms

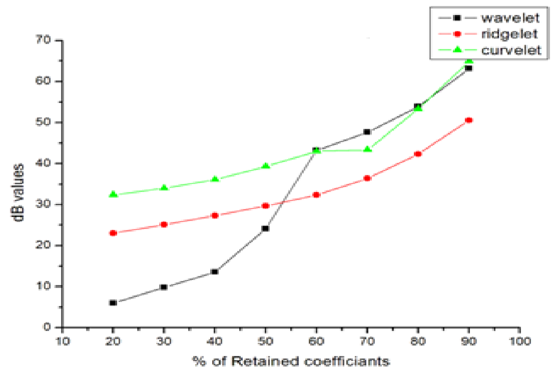

Fig.2:**PSNR of image F1**

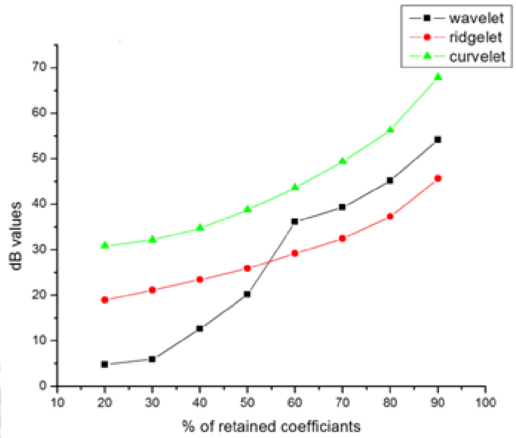

Fig.3:**PSNR of image F2**

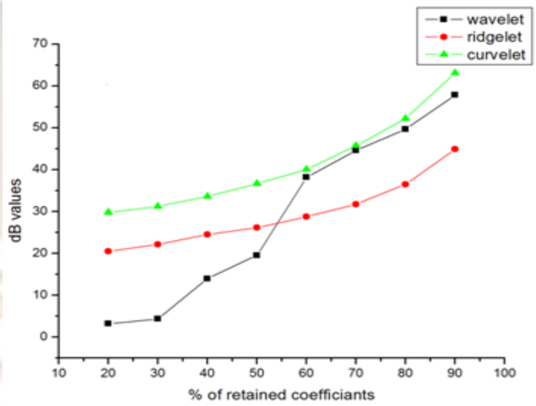

Fig.4:**PSNR of image F3**

Higher % of retained coefficients indicates lesser compression ratio.

Table 1: PSNR Values

(% RC)	WT (PSNR)	RT (PSNR)	CT (PSNR)
90	63.17	50.55	64.93
80	53.89	42.32	53.34
70	47.65	36.39	43.37
60	43.19	32.35	43.00
50	24.10	29.66	39.31
40	13.52	27.32	36.10
30	9.79	25.10	33.95
20	6.02	23.03	32.38

RC – Retained Coefficients,
WT- Wavelet Transform,
RT- Ridgelet Transform,
CT- Curvelet Transform,
PSNR- Peak Signal to Nose Ratio.

The algorithms applied for three images Rose.jpg of size 97×97, Lena.png of size 503×503 and Fruits.png of size 131×131.

Fig.5: Implementation Results

V. CONCLUSIONS

From the table it is clear that the proposed compression algorithm for color image using CT results in higher PSNR values compared to WT and RT at all compression ratios. The performance of WT at lower % RC is unacceptable since, WT has limitations in handling the line and curve singularities in the image. From table it is observed that the RT has improved results compared to WT, but the results are lower compared to CT, this is because the RT are designed to handle line singularities. The proposed compression technique is tested for natural imaged which have singularities in curves, the CT results in best compared to WT and RT. The CT performs better compared to WT and RT especially at lower % RC, which means that even though a large number of coefficients neglected the CT is capable of retrieving higher details compared to RT, which results in getting a quality image at all compression ratios.

References

[1] Sayood, Khalid (2000),” Introduction to Data Compression”, Second edition Morgan Kaufmann.

[2] David Salomon, *Data Compression, The Complete Reference*, (2nd Edition, 2001, Reprint, Springer Verlag).

[3] Gonzalez, Woods, Eddins, *Digital Image Processing using MATLAB* (Pearson Education, 2004).

[4] Gonzalez, Woods, Eddins, *Digital Image Processing* (Pearson Education, 2008).

[5] E.J. Candes, Ridgelets, Theory and Applications, Ph.D. Thesis, Aug.1998, Department of Statistics, Stanford University.

[6] Minh N. Do and Martin Vetterli, The Finite Ridgelet Transform for Image Representation, *IEEE Transactionson Image Processing*, January 2003.

[7] E.J. Candes, Laurent Demanet, D.L. Donoho, Lexing Ying, “Fast Discrete Curvelet Transforms”, Stanford University Press Notes, July 2005.

[8] F. Matus and J. Flusser, Image representation via a finite Radon transform, *IEEE Trans. Pattern Anal. MachineIntelligence*, vol. 15, no. 10, pp. Oct 1993.

[9] M. N. Do and M. Vetterli, The Contourlet transform: an efficient directional multi-resolution image representation, IEEE Trans. Image Processing, Vol 14, No.12, Dec 2005.

[10] AwaisMansoor and AtifBinMansoor, “ On Image Compression Using Digital Curvelet Transform”, 9th International Multitopic Conference, IEEE Dec. 2005.