Journal of Signal and Image Processing
ISSN: 0976-8882 & E-ISSN: 0976-8890, Vol. 2, Issue 1, 2011, pp-13-18
Available online at http://www.bioinfo.in/contents.php?id=48

PERFORMANCE ANALYSIS OF IMAGE COMPRESSION USING CURVELET TRANSFORM

KAMLESH GUPTA[1]*, SANJAY SILAKARI[2]
[1]Jaypee University of Engineering & Technology , Guna, MP, India, email- kamlesh_rjitbsf@yahoo.co.in
[2]University Institute of Technology, Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal, India, email- ssilakari@yahoo.co.in
*Corresponding author. E-mail: kamlesh_rjitbsf@yahoo.co.in

Received: June 28, 2011; Accepted: July 23, 2011

Abstract- This paper proposes a novel image compression algorithm using curvelet transform. The original image was decomposed into curvelet coefficients using fast discrete curvelet transform, after that the different scales of quantized curvelet coefficients were selected for lossy compression and arranged in descending order. Then we set the cutoff threshold for curvelet coefficients. The proposed method was compared with image compression method based on wavelet transform. Experimental results show that the compression performance of our method gains much improvement based on PSNR. Moreover, the algorithm works fairly well for declining block effect at higher compression ratios.

Key words— Image compression, wavelet transform, curvelet transform.

1. Introduction

From near the beginning days to now, the basic objective of image compression is the reduction of size for transmission or storage while maintaining suitable quality of reconstructed images. For this purpose many compression techniques i.e. scalar/vector quantization, differential encoding, predictive image coding, transform coding have been introduced. Among all these, transform coding is most efficient especially at low bit rate [1].

In the past few years, wavelets and related multi-scale representations pervade all areas of signal processing. The reason for the success of wavelets is the fact that wavelet bases represent well a large class of signals, and therefore allow us to detect roughly isotropic features occurring at all spatial scales and locations. However, there has been a growing awareness to the observation that wavelets may not be the best choice for presenting natural images recently. This observation is due to the fact that wavelets are blind to the smoothness along the edges commonly found in images. In other word, the wavelet can't provide the 'sparse' representation for an image

Hence, recently, some new transforms have been introduced to take advantage of this property. The E.J Candes and D.L Donoho was introduced in transform called curvelet transform. The curvelet transform is a special member of the multi-scale geometric transforms [2, 3, 4]. It is a transform with multi-scale pyramid with many directions at each length scale. Curvelets will be superior over wavelets in following cases:

- Optimally sparse representation of objects with edges.
- Optimal image reconstruction in severely will posed problems.
- Optimal sparse representation of wave propagators.

The curvelet transform based coding performance is better for Compression over Wavelet transform based coding for gray scale and color images. Even though the coefficients neglected are large, the higher PSNR values in curvelet case show that it has better reconstruction performance.

The rest of the paper is organized as follows. Section 2 provides a description of second generation continuous and digital curvelet transform. Section 3 proposed image compression algorithm. Section4 describe the performance analysis for image compression based on wavelet and curvelet. Conclusion and future work is described in section 5.

2. Second Generation of Curvelet Transforms

The curvelet transform has gone through two major revisions. The first curvelet transform [3] (commonly referred to as the “curvelet 99” transform now) used a complex series of steps involving the ridgelet analysis of the radon transform of an image. The performance was exceedingly slow. Soon after their introduction, researchers developed numerical algorithms for their implementation, and reported on a series of practical successes [5].

2.1 *Continuous-Time Curvelet Transforms*

In 2000, Candes and Donoho introduced the curvelet transform [6]. The continuous curvelet transform can be defined by a pair of windows $W(r)$ (a radial window) and $V(t)$ (an angular window), with variables W as a frequency-domain variable, and r and θ as polar coordinates in the frequency-domain.

$\sum_{j=-\infty}^{\infty} w^2 (2^j r) = 1, \quad r \in \left(\frac{3}{4}, \quad \frac{3}{2}\right)$............(1)

$\sum_{l=\infty}^{\infty} v^2 (t-1) = 1, \quad t \in \left(-\frac{1}{2}, \quad \frac{1}{2}\right)$............(2)

A polar 'wedge' represented by *Uj* is supported by *W* and *V*, the radial and angular windows. *Uj* is defined in the Fourier domain by

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

The curvelet transform can be defined as a function of *x* =(*x*1, *x*2) at scale2^{-j}, orientation θ_l, and position $x_k^{(j,l)}$by

$\theta_{j,l,k}(x) = \varphi_j\left(R_{\theta_t}\left(x - x_k^{(j,l)}\right)\right) \dots\dots\dots\dots(4)$

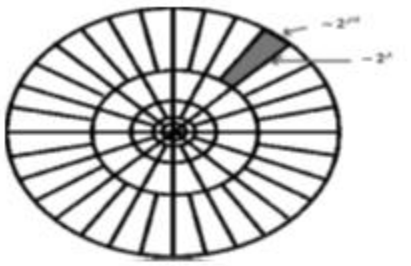


Fig. 1- Curvelet tiling in the frequency domain

Where R_θ is the rotation in radians? Fig.(1) illustrates the polar 'wedges' represented byU_j. Further details are presented in [6].

2.2 Digital curvelet transforms

In the continuous-time definition (3), the window Uj smoothly extracts frequencies near the dyadic corona and near the angle. Coronae and rotations are not especially adapted to Cartesian arrays. Instead, it is convenient to replace these concepts by Cartesian equivalents; here, "Cartesian coronae" based on concentric squares (instead of circles) and shears, as show in Fig. (2).

Define the "Cartesian" window

$\widetilde{U}_J(\omega) = \widetilde{W}_J(\omega) V_j(\omega) \dots\dots\dots..(5)$

$\widetilde{W}_J(\omega)$ is a window of the form

$\widetilde{W}_J(\omega) = \sqrt{\varphi_{j+1}^2(\omega) - \varphi_j^2(\omega)} \quad j \geq 0$ .... (6)

Where φ is defined as the product of low-pass one-dimensional windows

$\varphi_j(\omega_1, \omega_2) = \varphi(2^j \omega_1)\varphi(2^j \omega_2)$ .... (7)

The function φ obeys $0 \leq \varphi \leq 1$, might be equal to 1 on [-1/2, 1/2], and vanishes outside of [-2, 2]. The digital curvelet transform coefficient is obtained by

$c(j,l,k) = \int f(\dot{\omega}) \widetilde{U}_J \left(S_{\theta_l}^{-l}\omega\right) e^l \langle S_{\theta_l}^T b, \alpha x \rangle \, d\omega$ ... (8)

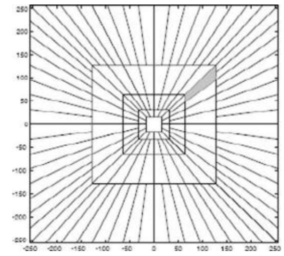


Fig. 2- Digital curvelet tiling of space and frequency.

3. Image compression

3.1 Image compression algorithm using Curvlet transforms

Step 1: calculate the cuvelet coefficient of the image planes using following equations

$c(j,l,k) = \int_{R^2} f(x) \overline{\psi_{j,l,k}(x)} \, dx$

Where R denote the real line.

Step 2: Calculate the size of compressed image according to given Compression ratio (CPR).

Step 3: Arrange the Curvelet coefficients in descending order C.

Step 4: Find out the cutoff threshold for Curvelet coefficients (CL) as given below

N = CPR*Image size

Cutoff = CL *(N)

Where CL is the curvelet coefficients array arranged in descending order.

Step 5: Remove all the coefficients below cutoff

C1 = C > Cutoff

Step 6: Perform inverse curvelet transform of C1 to get compressed image.

4. Performance and Result analysis

It is difficult to provide a performance analysis of the proposed new method and we will instead present some experimental results to illustrate its performance. The proposed algorithm is evaluated on 256×256 Lena color images and compared with traditional image compression methods based on wavelet transform, which pervade all areas of signal processing. As an objective measure of reconstructed image quality, the PSNR (peak signal to noise ratio) in decibels is used [7] and is defined as

$$PSNR = 10 \log_{10} \frac{255^2}{MSE}$$

Where

$$MSE = \frac{\sum_{i-1}^{W} \sum_{j-i}^{H} (x_{ij} - \widetilde{x_{ij}})}{W \times H}$$

Where x_{ij} and $\widetilde{x_{ij}}$ denote the original and reconstructed pixel, respectively, and the images are of size W× H.

5. Conclusion and future work

In this paper, we have proposed a new color image compression algorithm based on second generation of curvelet transforms. We set the cutoff threshold of image compression for curvelet coefficients. Experimental results show that curvelet transform based coding performance is better for compression over JPEG and Wavelet transform based coding for grayscale and color images. Even though the coefficients neglected are large, the higher PSNR values in curvelet case show that it has better reconstruction performance. Use of proper threshold method improves the performance still further. This motivated the study further for curvelet transform.

References

[1] Muhammad Azhar Iqbal, Dr Muhammad Younus Javed, Usman Qayyum, "Curvelet-based Image Compression with SPIHT" (2007) *IEEE computer society, International*

Conference on convergence Information Technology.

[2] Candes E.J., Laurent Demanet, Donoho D.L., Lexing Ying (2005) *"Fast Discrete Curvelet Transforms", Stanford University Press Notes.*

[3] Cand`es E. J. and Donoho D. L. (2000) *Curvelets, "a surprisingly editors, effective non-adaptive representation for objects with edges", In C. Rabut A. Cohen and L. L. Schumaker, Curves and Surfaces, 105–120, Vanderbilt University Press.*

[4] Cand`es E. J. and Donoho D. L. (2004) *Comm. on Pure and Appl. Math*. 57, pp.219–26.

[5] Starck J. L., Murtagh F., Candès E. J. & Donoho D. L. (2003) *IEEE Transactions on Image Processing*,12(6), 706–717.

[6] Cand`es E. J. and Donoho D. L. (2000) Wavelet Applications in Signal and Image Processing, vol. VIII, 4119-01, SPI.

[7] Granrath D. J. (1981) *"The role of human visual models in image processing", Proceedings of the IEEE*, 69, 552–561.

3.2 Flowchart for image compression using Curvlet transforms:

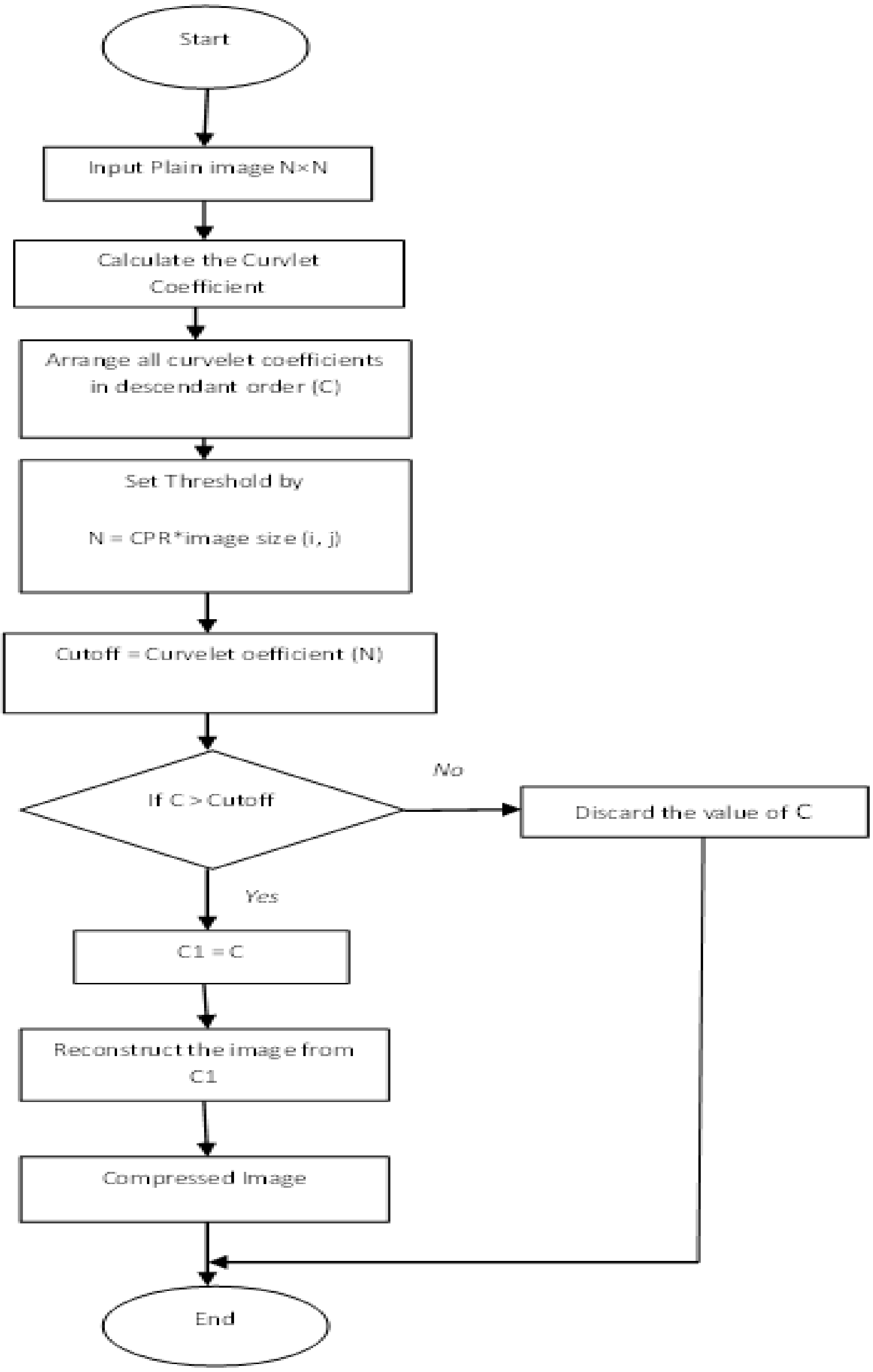


Fig. 3-Flowchart for image compression using Curvelet

Table 1- Comparison of PSNR for different compression ratio using wavelet and curvelet.

Lena 256×256 (PSNR)

Compression Ratio	1:20	1:30	1:40	1:50	1:60	1:70	1:80	1:90	1:100
Curvelet	45.18	38.68	31.93	24.97	21.82	19.83	18.42	17.29	16.38
Wavelet	38.63	28.25	21.61	18.08	15.82	14.38	13.83	12.66	12.11

Lena 256×256 (MSE)

Compression Ratio	1:20	1:30	1:40	1:50	1:60	1:70	1:80	1:90	1:100
Curvelet	2.94	9.44	16.19	23.15	26.30	28.29	29.71	30.83	31.74
Wavelet	9.49	19.88	26.51	30.04	32.30	33.74	34.74	35.47	36.02

Lena 256×256 (Time in Second)

Compression Ratio	1:20	1:30	1:40	1:50	1:60	1:70	1:80	1:90	1:100
Curvelet	1.423	1.391	1.298	1.403	1.169	1.241	1.277	1.600	1.292
Wavelet	0.069	0.180	0.172	0.228	0.177	0.172	0.151	0.152	0.245

Baboon 256×256 (PSNR)

Compression Ratio	1:20	1:30	1:40	1:50	1:60	1:70	1:80	1:90	1:100
Curvelet	42.24	34.98	29.16	22.78	19.16	17.08	15.69	14.69	13.94
Wavelet	34.69	26.93	18.74	15.47	13.69	12.53	11.71	11.08	10.59

Baboon 256×256 (MSE)

Compression Ratio	1:20	1:30	1:40	1:50	1:60	1:70	1:80	1:90	1:100
Curvelet	5.88	13.14	18.97	25.34	28.97	31.05	32.43	33.43	34.19
Wavelet	13.43	21.20	29.39	32.65	34.43	35.60	36.42	37.04	37.53

Baboon 256×256 (Time in Second)

Compression Ratio	1:20	1:30	1:40	1:50	1:60	1:70	1:80	1:90	1:100
Curvelet	1.244	1.649	1.431	1.355	1.521	1.287	1.421	1.421	1.530
Wavelet	0.143	0.166	0.169	0.181	0.168	0.112	0.162	0.310	0.190

Pepper 256×256 (PSNR)

Compression Ratio	1:20	1:30	1:40	1:50	1:60	1:70	1:80	1:90	1:100
Curvelet	48.03	39.69	32.53	25,11	20.99	18.47	16.80	15.60	1.71
Wavelet	43.08	32.32	19.87	16.36	14.43	13.04	11.99	11.19	10.57

Pepper 256×256 (MSE)

Compression Ratio	1:20	1:30	1:40	1:50	1:60	1:70	1:80	1:90	1:100
Curvelet	0.12	8.43	15.59	23.01	27.13	29.65	31.32	32.52	33.42
Wavelet	5.04	15.80	28.25	31.76	33.69	35.08	36.13	36.93	37.55

Pepper 256×256 (Time in Second)

Compression Ratio	1:20	1:30	1:40	1:50	1:60	1:70	1:80	1:90	1:100
Curvelet	1.437	1.346	1.920	2.042	1.612	1.336	1.519	1.463	14.71
Wavelet	0.167	0.103	0.189	0.153	0.123	0.158	0.173	0.232	0.070

Airplane 256×256 (PSNR)

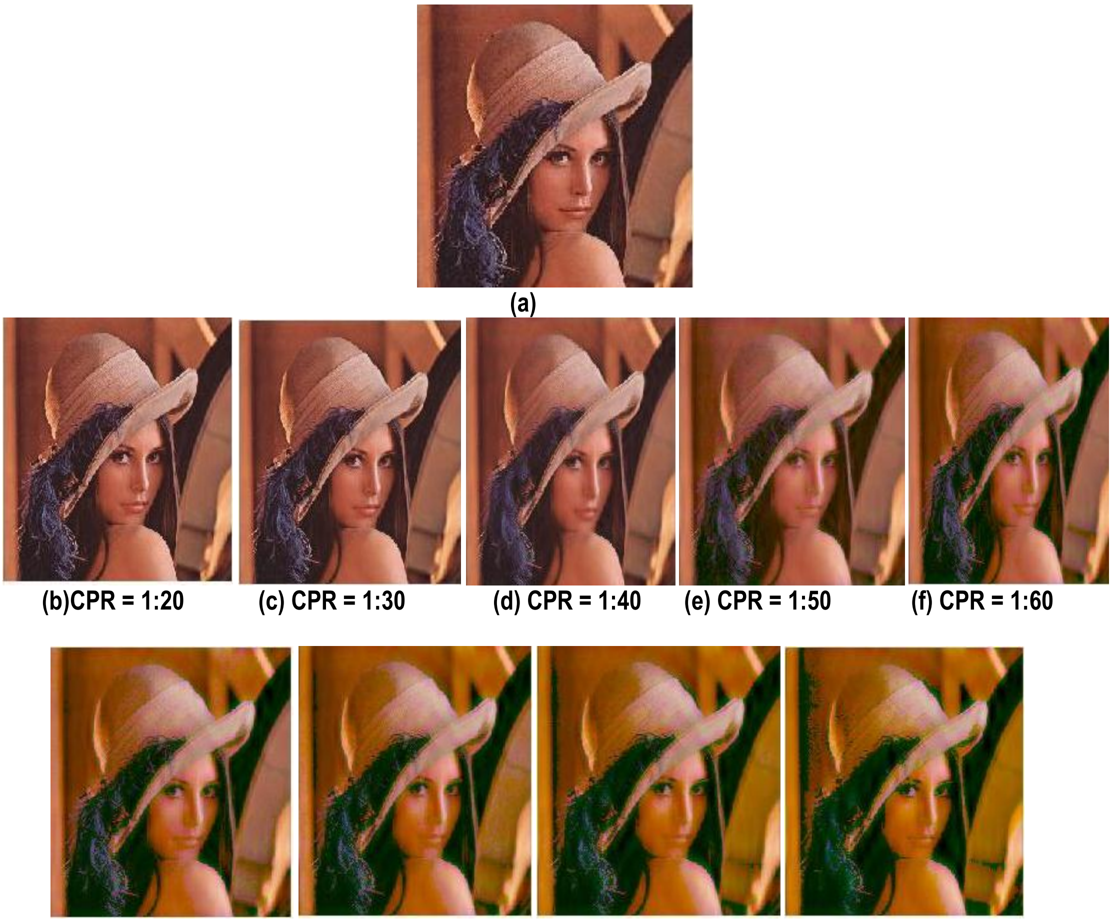
Compression Ratio	1:20	1:30	1:40	1:50	1:60	1:70	1:80	1:90	1:100
Curvelet	49.06	40.13	32.24	24.67	20.87	18.01	16.07	14.78	13.97
Wavelet	43.93	30.87	20.29	16.13	14.14	12.96	11.74	11.31	10.72

Airplane 256×256 (MSE)

Compression Ratio	1:20	1:30	1:40	1:50	1:60	1:70	1:80	1:90	1:100
Curvelet	-0.92	8.00	15.88	23.46	27.25	30.11	32.05	33.34	34.15
Wavelet	04.19	17.25	27.83	31.99	33.99	35.16	36.38	36.81	37.40

Airplane 256×256 (Time in Second)

Compression Ratio	1:20	1:30	1:40	1:50	1:60	1:70	1:80	1:90	1:100
Curvelet	1.656	1.952	1.851	1.665	1.582	1.499	1.633	1.318	1.583
Wavelet	0.201	0.107	0.159	0.06	0.190	0.167	0.149	0.146	0.109



(a)

(b)CPR = 1:20 (c) CPR = 1:30 (d) CPR = 1:40 (e) CPR = 1:50 (f) CPR = 1:60

(g) CPR = 1:70 (h) CPR = 180 (i) CPR = 1:90 (j) CPR = 1:100

Fig. 4 (b-j) Reconstructed Lena Image for different compression ratio in Curvelet transform and fig.4 (a) Original Lena image 256×256.

(b) CPR = 1:20 (c) CPR = 1:30 (d) CPR = 1:40 (e) CPR = 1:50 (f) CPR = 1:60

(g) CPR = 1:70 (h) CPR = 180 (i) CPR = 1:90 (j) CPR = 1:100

Fig. 5 (b-j) Reconstructed Lena Image for different compression ratio in Wavelet transform.