International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.8, No.7 (2015), pp.199-212
http://dx.doi.org/10.14257/ijsip.2015.8.7.19

# Novel Approach for Image Compression Using Curvelet Transform

Kamlesh Gupta[1] and Ranu Gupta[2]

*[1] RustamJi Institute of Technology, Boarder Security Force, Tekanpur, MP, India*
*[2] Jaypee University of Engg. & Technology, Guna, MP, India*
*kamlesh.gupta@juet.ac.in, ranu.gupta@juet.ac.in*

***Abstract***

*This paper proposes a novel image compression algorithm using curvelet transform. The proposed research decomposed the original image into curvelet coefficients by using fast discrete curvelet transform, after that the different scales of quantized curvelet coefficients were selected for lossy compression by the help of cutoff threshold. 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, MSE and memory size. Moreover, the algorithm works fairly well for declining block effect at higher compression ratios.*

***Keywords:*** *Curvelet Transform, Wavelet Transform, Peak Signal to Noise Ratio and Mean Square Errors*

## 1. Introduction

The basic objective of image compression is the reduction of size for transmission or storage while maintaining suitable quality of reconstructed images. Since the lossy schemes can produce much higher compression ratios than the lossless ones, for this purpose many compression techniques have been introduced in [1]. Most of them are efficient for low bit rates.

In the past few years, wavelets [2, 3] 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, so the second generation curvelet transform are use for image compression.

## 2. Curvelet Transform

The E. J Candes and D.L Donoho were introduced new transform called curvelet transform. The curvelet transform is a special member of the multi-scale geometric transforms [5-7]. 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.
- Reduce redundancy smartly from the image.

ISSN: 2005-4254 IJSIP

The curvelet transform based coding performance is better for compression over wavelet transform based coding for gray scale and color images. If the coefficient neglect large, the curvelet transform has better reconstruction over wavelet transform and remove the redundancy from the image, so it is beneficial for image encryption.

### 2.1 Continuous-Time Curvelet Transforms

In 2000, Candes and Donoho introduced the curvelet transform [7, 4]. 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.

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

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

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

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

The curvelet transform can be defined as a function of $x$ = (x1, x2) at scale $2^{-j}$, orientation $\theta_l$, and position $x_k^{(j,l)}$by

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

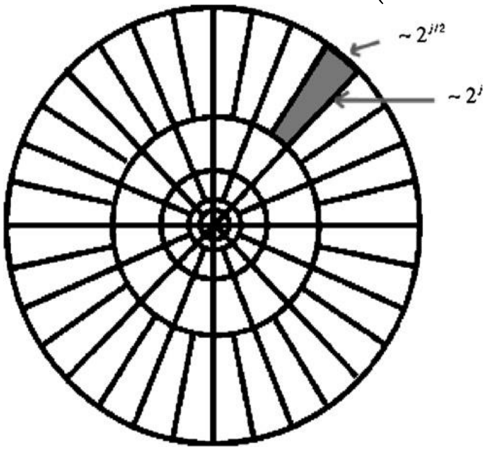



**Figure 1. Curvelet Tiling in the Frequency Domain**

Where $R_\theta$ is the rotation in radians. Figure 1 illustrates the polar 'wedges' represented by $U_j$. Further details are presented in [5].

### 2.2 Digital Curvelet Transforms

In the continuous time definition (2.1), the window $U_j$ 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 shown in figure 2. Define the ''Cartesian" window

$$\widetilde{U_J}(\omega) = \widetilde{W_J}(\omega) V_j(\omega) \dots\dots\dots\dots\dots\dots\dots\dots \qquad \mathbf{(5)}$$

$\widetilde{W_J}(\omega)$ is a window of the form.

$$\widetilde{W_J}(\omega) = \sqrt{\varphi_{j+1}^2(\omega) - \varphi_j^2(\omega)} \quad J \geq 0 \dots \qquad \mathbf{(6)}$$

Where $\varphi$ is defined as the product of low-pass one-dimensional windows.

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

The function $\varphi$ 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

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

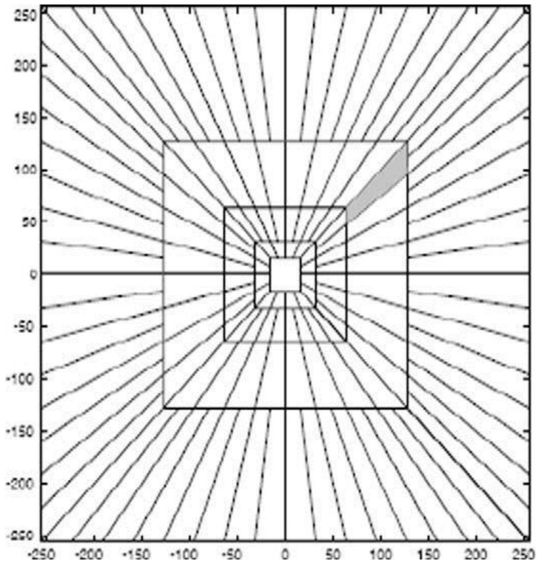



**Figure 2. Digital Curvelet Tiling of Space and Frequency**

## 3. Flowchart for Image Compression Based On Curvelet Transforms

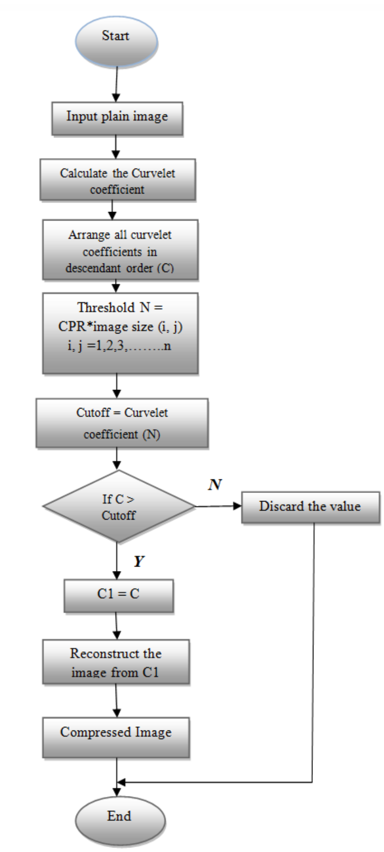



**Figure 3. Curvelet based Image Compression**

## 4. Algorithm for Image Compression Based On Curvelet Transforms

This research 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 and MSE. Moreover, the algorithm works fairly well for declining block effect at higher compression ratios.

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

$$c(j,l,k) = \int_{R^2} f(x)\overline{\psi_{j,l,k}(x)}\,dx \qquad \dots\dots\dots \quad (\mathbf{9})$$

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

$$N = CPR * Image\ Size \dots\dots\dots.. \qquad (\mathbf{10})$$

$$Cutoff = CL * (N) \dots\dots\dots\dots\dots.. \qquad (\mathbf{11})$$

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

**Step 5**: Remove all the coefficients below cutoff

$$C1 = c > Cutoff \ \dots\dots\dots\dots\dots\dots \qquad (\mathbf{12})$$

C1 = C > Cutoff

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

## 5. Simulation Result

### 5.1 PSNR and MSE for Image Compression

The proposed algorithm is evaluated on 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 by the PSNR (Peak Signal to Noise Ratio) and MSE (Mean Square Error).

A lower value for MSE means lesser error, and as seen from the inverse relation between the MSE and PSNR, this translates to a high value of PSNR. Logically, a higher value of PSNR is good because it means that the ratio of Signal to Noise is higher. Here, the 'signal' is the original image, and the 'noise' is the error in reconstruction. So, a compression scheme having a lower MSE (and a high PSNR), can be recognized as a better one. We evaluated the performance of image compression with an un-optimized MATLAB 7.0 code. Performance was measured on a machine with Intel Core 2 Duo 2.00 GHz CPU with 2 GB of RAM running on Windows XP.

Table 1, 2 and 3 shows the simulation results of various standard images (from Signal Image Processing Institute SIPI) of different size. In these tables we simulate the result on 120 × 120, 256× 256, and 512 × 512 standard 24-bit color, 96 dpi JPEG images. It is clear that the PSNR and MSE value on 120 × 120 in table 1 and figure 4, 5 and 256 × 256 in table 2 and figure 6, 7 shows that the image reconstruction quality for curvelet transform is better than that of wavelet transform. The figure 4, 5, 6, 7, 8 and 9 indicate that curvelet show better PSNR and MSE for 120 ×120, 256×256 and 512×512 Lena 24-bit color image compare to wavelet transform respectively. The higher PSNR and lower MSE value show the better reconstruction quality of the image. The image data are highly redundant, so in the case of curvelet the redundancy of the image is highly reduced which in turn gives the advantage in the image encryption.

**Table 1. PSNR and MSE for Different Compression Ratio of Different Color Image**

| Lena 120×120 (PSNR) | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Compression Ratio** | **1:20** | **1:30** | **1:40** | **1:50** | **1:60** | **1:70** | **1:80** | **1:90** | **1:100** |
| **Curvelet** | 40.01 | 32.41 | 24.03 | 14.79 | 11.46 | 9.06 | 8.34 | 7.38 | 6.64 |
| **Wavelet** | 37.87 | 29.31 | 22.26 | 12.44 | 8.97 | 7.12 | 5.96 | 5.16 | 4.57 |
| **Lena 120×120 (MSE)** | | | | | | | | | |
| **Compression Ratio** | **1:20** | **1:30** | **1:40** | **1:50** | **1:60** | **1:70** | **1:80** | **1:90** | **1:100** |
| **Curvelet** | 8.11 | 15.71` | 24.03 | 33.33 | 36.67 | 38.52 | 39.78 | 40.74 | 41.48 |
| **Wavelet** | 10.25 | 18.81 | 25.86 | 35.68 | 39.15 | 41.00 | 42.16 | 42.96 | 43.55 |
| **Baboon 120×120 (PSNR)** | | | | | | | | | |
| **Compression Ratio** | **1:20** | **1:30** | **1:40** | **1:50** | **1:60** | **1:70** | **1:80** | **1:90** | **1:100** |
| **Curvelet** | 35.99 | 29.54 | 23.93 | 16.00 | 12.60 | 10.73 | 9.48 | 8.56 | 7.87 |
| **Wavelet** | 32.16 | 25.90 | 21.37 | 12.62 | 9.56 | 7.90 | 6.84 | 6.08 | 5.50 |
| **Baboon 120×120 (MSE)** | | | | | | | | | |
| **Compression Ratio** | **1:20** | **1:30** | **1:40** | **1:50** | **1:60** | **1:70** | **1:80** | **1:90** | **1:100** |
| **Curvelet** | 12.13 | 18.58 | 24.19 | 32.12 | 35.52 | 37.39 | 38.64 | 39.56 | 40.25 |
| **Wavelet** | 15.96 | 22.23 | 26.75 | 35.50 | 38.56 | 40.22 | 41.28 | 42.04 | 42.62 |
| **Pepper 120×120 (PSNR)** | | | | | | | | | |
| **Compression Ratio** | **1:20** | **1:30** | **1:40** | **1:50** | **1:60** | **1:70** | **1:80** | **1:90** | **1:100** |
| **Curvelet** | 38.85 | 31.59 | 25.23 | 19.14 | 15.42 | 13.07 | 11.47 | 10.35 | 9.49 |
| **Wavelet** | 37.25 | 29.30 | 23.18 | 14.58 | 11.32 | 9.47 | 8.12 | 7.12 | 6.35 |
| **Pepper 120×120 (MSE)** | | | | | | | | | |
| **Compression Ratio** | **1:20** | **1:30** | **1:40** | **1:50** | **1:60** | **1:70** | **1:80** | **1:90** | **1:100** |
| **Curvelet** | 9.27 | 16.53 | 22.89 | 28.99 | 32.70 | 35.05 | 36.65 | 37.77 | 38.63 |
| **Wavelet** | 10.87 | 18.82 | 24.94 | 33.54 | 36.80 | 38.65 | 40.00 | 41.01 | 41.77 |

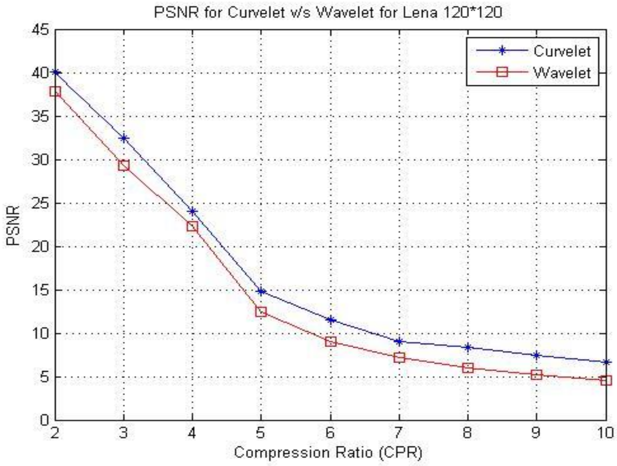



**Figure 4. MSE of Curvelet vs. Wavelet for Different Compression Ratio**

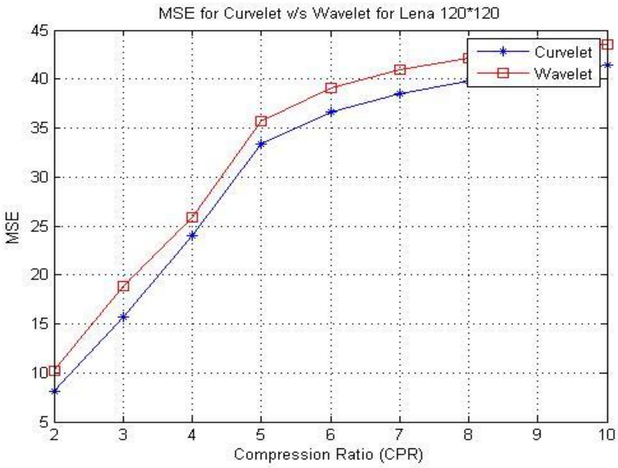



**Figure 5. MSE of Curvelet vs. Wavelet for Different Compression Ratio**

**Table 2. PSNR and MSE for Different Compression Ratio of Different Color Image**

| 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 |
| **Baboon 256×256 (PSNR)** | | | | | | | | | |
| **Compression** | **1:20** | **1:30** | **1:40** | **1:50** | **1:60** | **1:70** | **1:80** | **1:90** | **1:10** |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Ratio** | | | | | | | | | **0** |
| **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 |
| **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 |
| **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 |

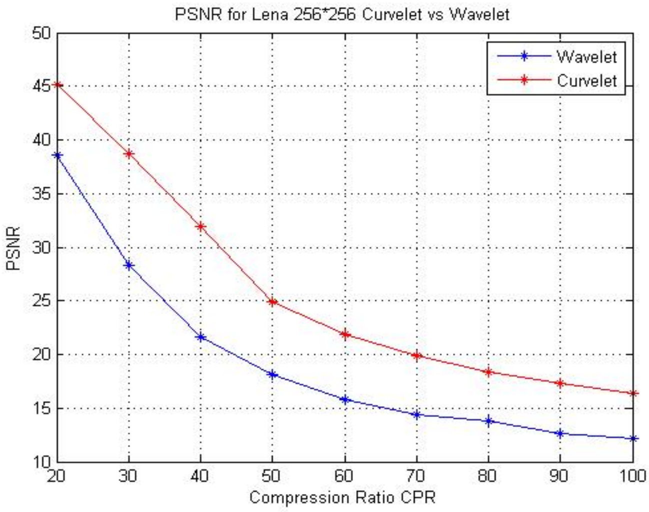



**Figure 6. PSNR of Curvelet vs. Wavelet for Different Compression Ratio**

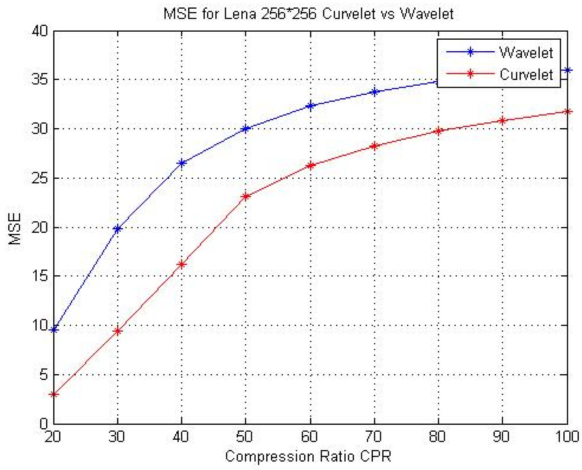



**Figure 7. MSE of Curvelet vs. Wavelet for Different Compression Ratio**

When the size of the image increases the results are contradictory for some images like baboon 512×512 and Peeper 512×512 as shown in table 3 and figures 10, 11, 12 and 13 i.e. the wavelet transform gives higher PSNR and lower MSE for baboon 512×512 image for the compression ratio 1:20, 1:30 and 1:40 as shown in figures 5.10 and 5.11 i.e. for low compression ratio PSNR are 41.80, 38.43 and 27.64 respectively but in the case of curvelet transform it is 40.56, 32.90, 26.39 and the MSE are 6.32, 13.29, and 20.48 but in curvelet it is 7.56, 15.22 and 21.73. As the compression ratio increases such as 1:50 or above the curvelet give higher PSNR (18.13, 14.29, 12.16, 10.75, 9.75, 8.99) and lower MSE over wavelet PSNR (16.66, 11.00, 9.07, 7.84, 6.97, 6.31) and MSE as shown in Figures 10 and 11.

**Table 3. PSNR and MSE for Different Compression Ratio of Different Color Image**

| **Lena 512×512 (PSNR)** | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Compression Ratio** | **1:20** | **1:30** | **1:40** | **1:50** | **1:60** | **1:70** | **1:80** | **1:90** | **1:100** |
| **Curvelet** | 47.24 | 39.40 | 30.59 | 15.74 | 12.18 | 10.25 | 8.97 | 8.02 | 7.26 |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Wavelet** | 45.23 | 36.81 | 27.71 | 13.17 | 9.35 | 7.35 | 6.13 | 5.28 | 4.66 |
| **Lena 512×512 (MSE)** | | | | | | | | | |
| **Compression Ratio** | **1:20** | **1:30** | **1:40** | **1:50** | **1:60** | **1:70** | **1:80** | **1:90** | **1:100** |
| **Curvelet** | 0.88 | 8.72 | 17.53 | 32.38 | 35.95 | 37.87 | 39.15 | 40.11 | 40.86 |
| **Wavelet** | 2.89 | 11.32 | 20.41 | 34.95 | 38.78 | 40.77 | 41.99 | 42.84 | 43.46 |
| **Baboon 512×512 (PSNR)** | | | | | | | | | |
| **Compression Ratio** | **1:20** | **1:30** | **1:40** | **1:50** | **1:60** | **1:70** | **1:80** | **1:90** | **1:100** |
| **Curvelet** | **40.56** | **32.90** | **26.39** | 18.13 | 14.29 | 12.16 | 10.75 | 9.75 | 8.99 |
| **Wavelet** | **41.80** | **34.83** | **27.64** | 14.66 | 11.00 | 9.07 | 7.84 | 6.97 | 6.31 |
| **Baboon 512×512 (MSE)** | | | | | | | | | |
| **Compression Ratio** | **1:20** | **1:30** | **1:40** | **1:50** | **1:60** | **1:70** | **1:80** | **1:90** | **1:100** |
| **Curvelet** | **7.56** | **15.22** | **21.73** | 29.99 | 33.83 | 35.97 | 37.37 | 38.37 | 39.13 |
| **Wavelet** | **6.32** | **13.29** | **20.48** | 33.46 | 37.12 | 39.05 | 40.28 | 41.15 | 41.81 |
| **Pepper 512×512 (PSNR)** | | | | | | | | | |
| **Compression Ratio** | **1:20** | **1:30** | **1:40** | **1:50** | **1:60** | **1:70** | **1:80** | **1:90** | **1:100** |
| **Curvelet** | **47.19** | **38.34** | 31.19 | 21.84 | 17.01 | 14.11 | 12.31 | 11.04 | 10.10 |
| **Wavelet** | **47.62** | **39.16** | 30.20 | 15.54 | 11.79 | 9.80 | 8.33 | 7.24 | 6.43 |
| **Pepper 512×512 (MSE)** | | | | | | | | | |
| **Compression Ratio** | **1:20** | **1:30** | **1:40** | **1:50** | **1:60** | **1:70** | **1:80** | **1:90** | **1:100** |
| **Curvelet** | **0.93** | **9.78** | 16.93 | 26.28 | 31.12 | 34.01 | 35.81 | 37.08 | 38.02 |
| **Wavelet** | **0.50** | **8.96** | 17.92 | 32.58 | 36.33 | 38.32 | 39.79 | 40.88 | 41.69 |

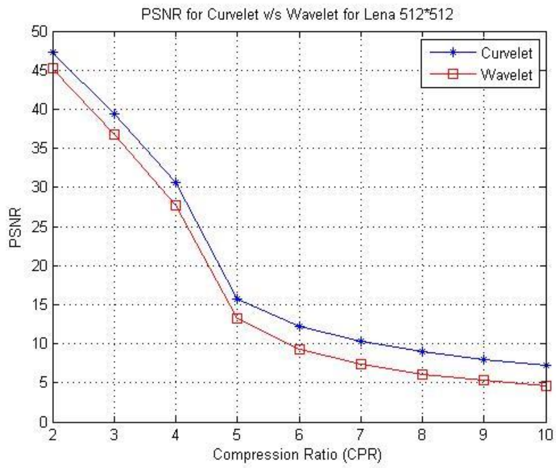



**Figure 8. PSNR of Curvelet vs. Wavelet for Different Compression Ratio**

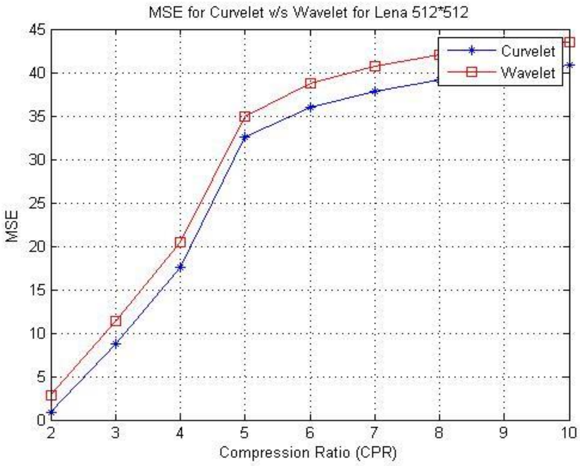



**Figure 9. MSE of Curvelet vs. Wavelet for Different Compression Ratio**

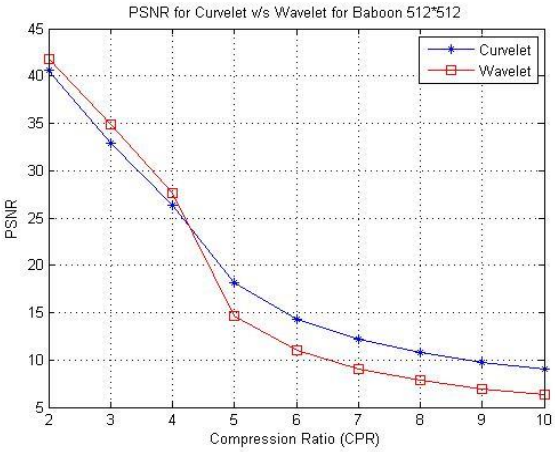



**Figure 10. PSNR of Curvelet vs. Wavelet for Different Compression Ratio**

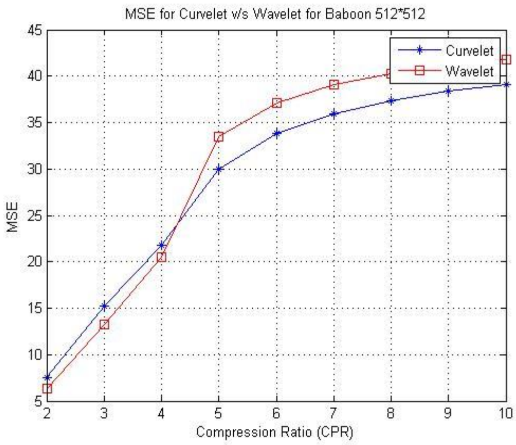



**Figure 11. MSE of Curvelet vs. Wavelet for Different Compression Ratio**

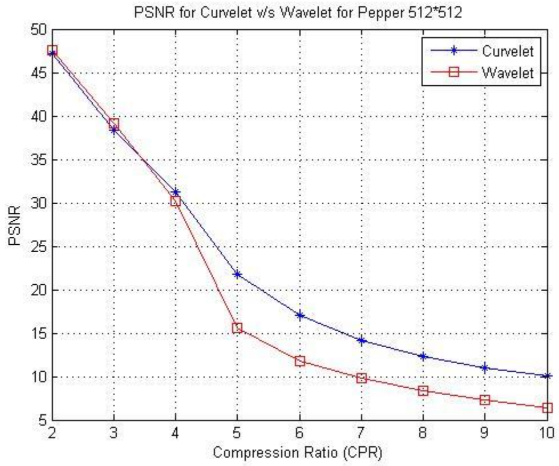



**Figure 12. PSNR of Curvelet vs. Wavelet for Different Compression Ratio**

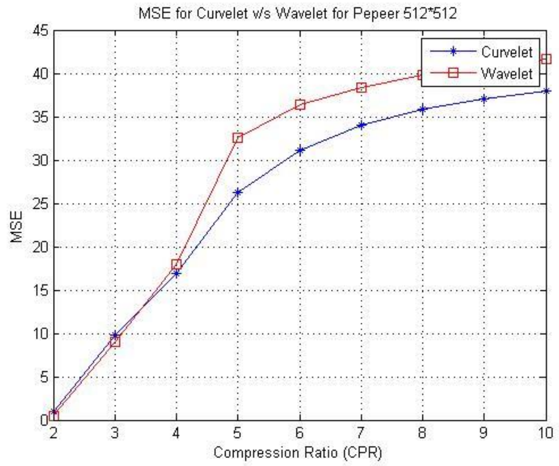



**Figure 13. MSE of Curvelet vs. Wavelet for Different Compression Ratio**

**Table 5. Image Compressed in Byte for Different Compression Ratio**

| | 1:20 | 1:30 | 1:40 | 1:50 | 1:60 | 1:70 | 1:80 | 1:90 | 1:100 |
|---|---|---|---|---|---|---|---|---|---|
| | **Lena 120×120 Plain image size 43200 bytes** | | | | | | | | |
| **Curvelet Transform** | 21600 | 14400 | 10800 | 8640 | 7200 | 6171 | 5400 | 4800 | 4320 |
| **Wavelet Transform** | 24221 | 17009 | 13415 | 11261 | 9812 | 8789 | 8021 | 7415 | 6932 |
| **Lena 256×256 Plain image size 196608 bytes** | | | | | | | | | |
| **Curvelet Transform** | 98304 | 65536 | 49152 | 39321 | 32768 | 28087 | 24576 | 21845 | 19660 |
| **Wavelet Transform** | 102589 | 69997 | 53632 | 43822 | 37255 | 32575 | 29080 | 26314 | 24124 |
| **Lena 512×512 Plain image size 786432 byte** | | | | | | | | | |
| **Curvelet Transform** | 393216 | 262144 | 196608 | 157286 | 131072 | 112347 | 98304 | 87381 | 78643 |
| **Wavelet Transform** | 465458 | 336836 | 271655 | 232301 | 206003 | 187289 | 173342 | 162233 | 153356 |

The compression and decompression time is measured for different size of color image by using curvelet transform is depicted in table 5.

The size of the compressed image are more in wavelet case as compared to curvelet as shown in table 5 and figures 14, 15 and 16, so its take less time for encryption and less bandwidth for transmission.

**Table 4. Time for Compression / Decompre-ssion Time Using Curvelet Transform**

| Lena (24-bit color image) | Comp Time in sec. | Decomp Time in sec. |
|---|---|---|
| 102×102 | 0.436 | 0.416 |
| 120×120 | 0.452 | 0.367 |
| 180×180 | 0.889 | 0.689 |
| 204×204 | 1.156 | 1.018 |
| 256×256 | 1.332 | 1.930 |
| 512×512 | 5.716 | 4.127 |

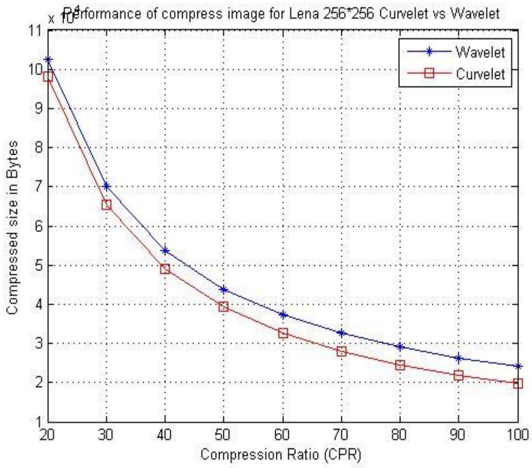



**Figure 15. Performance of Compress Image for Lena 256×256 Curvelet vs Wavelet**

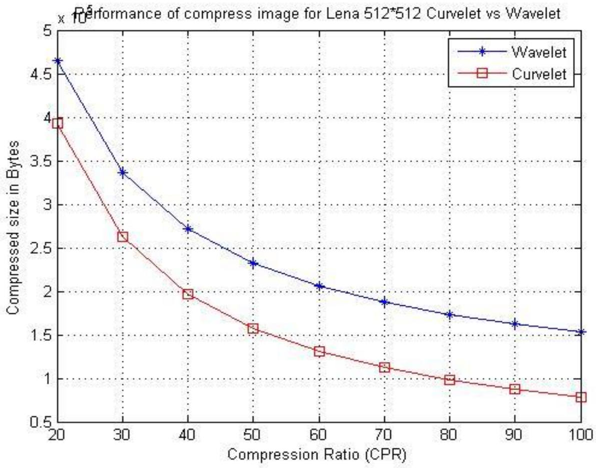



**Figure 16. Performance of Compress Image for Lena 512×512 Curvelet vs Wavelet**

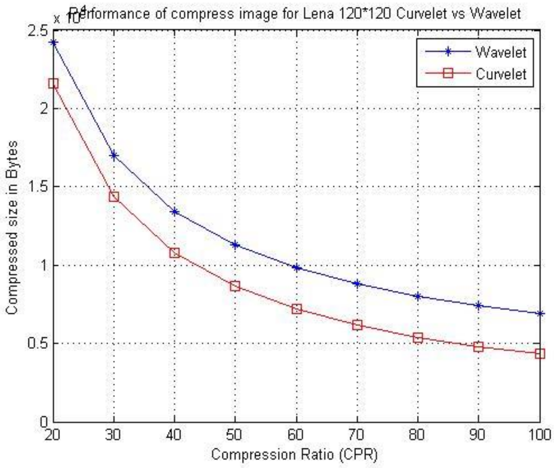
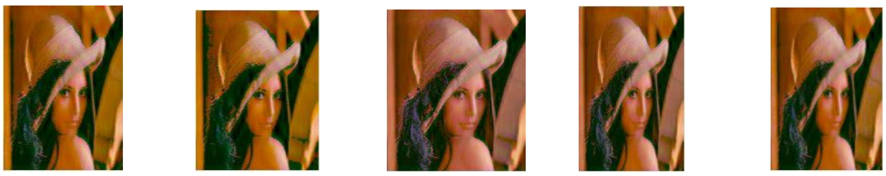



**Figure 14. Performance of Compress Image for Lena 120×120 Curvelet vs Wavelet**

The Figure 17(a) show the original 256×256, 24-bit JPEG Lena image. The figure 17 (b-j) and figure 18(b-j) shows the reconstruction quality of the compressed image by using curvelet transform and wavelet transform respectively with different compression ratio. In this case curvelet transform base coding give better reconstruction quality or low error for higher compression ratio compare to wavelet transform base coding.

(a) (b) 1:20 (c) 1:30 (d) 1:40 (e) 1:50



(f) 1:60 (g) 1:70 (h) 1:80 (i) 1:90 (j) 1:100

**Figure 17. (b-j) Reconstructed Lena Image for Different Compression Ratio in Curvelet Transform and (a) Original Lena image 256×256**

(b) 1:20 (c) 1:30 (d) 1:40 (e) 1:50

**(f) 1:60 (g) 1:70 (h) 1:80 (i) 1:90 (j) 1:100**

**Figure 18. (b-j) Reconstructed Lena Image for Different Compression Ratio in Wavelet Transforms**

## 6. Summary

In this paper, with the help of simulation results it is clear that curvelet transform gives the better performance for PSNR and MSE over the wavelet transform. More over the curvelet transforms are more suitable for the image data to represent the singularities over geometric structures in the image, than the wavelet counterpart because cuevelet are designed to handle the singularities on the curves, where as wavelets are effective for point singularities; hence curvelet transforms can be effectively used for the image compression with quality reconstruction. Thus curvelet based image compression, eliminate the redundancy from the image and achieve higher compression that reduces the encryption time.

## References

[1] G. V. Tcheslavski, “Image compression Fundamentals”, ELEN 4304/5365 DIP, Spring (**2008**), pp. 1-15.

[2] E. Le Pennec and S. Mallat “Sparse geometrical image approximation with bandlets”, IEEE Trans. Image Processing, vol. 14, no. 4, (**2005**), pp. 423-438.

[3] M. Do and M. Vetterli “The contourlet transform: an ancient directional multiresolution image representation”, IEEE Transaction Image Processing, vol. 14, no. 12, (**2005**), pp. 2091-2106.

[4] E. Candes, and D. Donoho “Continuous curvelet transform-II Discretization and frames”, Applied computation Harmon. Anal., vol. 19, (**2003**), pp. 198-222.

[5] E. Candes, L. Demanet, D. Donoho and L. Ying “Fast discrete curvelet transforms”, Multi-scale Model. Simul., vol. 5, no. 3, (**2006**), pp. 861-899.

[6] E. Candes and D. Donoho, “New tight frames of curvelets and optimal representations of objects with piecewise singularities”, Comm. Pure Appl. Math., vol. 57, (**2004**), pp. 219-266.

[7] E. Candes and D. Donoho “A surprisingly effective non-adaptive representation for objects with edges”, Vanderbilt Univ. Press, Nashville, (**2000**), pp. 105-120.

## Authors

**Kamlesh Gupta**, he is an Associate professor in Department of Information Technology, RustamJi Institute of Technology, Boarder Security Force, Gwalior, M.P., India. He has published 15 papers in International and National journals and conferences. His research interest includes cryptography and image processing. He is a life member of IETE.

**Ranu Gupta**, he Assistant professor in Department of Electronics and Engg. Jaypee University of Engg. & Technology, Guna, MP, India. He has published 5 papers in International and National journals and conferences. His research interest includes cryptography and image processing.