Image Denoising Using Curvelet Transform

Priti Naik and Shalini Bhatia

Abstract--**An image is often corrupted by noise in its acquisition and transmission. Hence, noise reduction is a required step for any sophisticated image processing algorithm. Denoising or noise reduction has been permanent research topic for engineers and scientists and one reason for it is the lack of a single technique, which is able to achieve denoising for a wide class of images. Though, traditional linear noise removal techniques like Wiener filtering have been existing for a long time for their simplicity and are able to achieve significant noise removal when the variance of noise is low, they cause blurring and smoothing of the sharp edges of the image . This paper includes two implementations of curvelet transform, curvelet transform via wrapping and curvelet transform via USFFT using two filtering methods such as hard threshold and partial reconstruction.**

Index Terms-- **Curvelets,, FFT, Filtering, Thresholding rules, USFFT, Wavelets**

I. Introduction

The need for efficient image restoration methods has grown with the massive production of digital images and movies of all kinds, often taken in poor conditions. No matter how good a camera is, an image improvement is always desirable to extend its range of action. Two main limitations in image accuracy are blur and noise. Blur is intrinsic to image acquisition systems, as digital images have finite number of samples and must satisfy the Shannon Nyquist sampling condition. The second main image perturbation is noise.

Image denoising is an important image processing task, both as a process itself, and as a component in other processes. Many ways to denoise an image or a set of data exist. The main properties of a good image denoising model are that it will remove noise while preserving edges. Traditionally, linear models have been used. One common approach is to use a Gaussian filter, or equivalently solve the heat-equation with the noisy image as input-data, i.e. a linear, 2nd order PDE-model. For some purposes this kind of denoising is adequate. One major advantage of linear noise removal models is the speed. But a drawback of the linear models is that they are not able to preserve edges in a good manner: edges, which are recognized as discontinuities in the image, are smeared out. Nonlinear models on the other hand can handle edges in a much better way than linear models can.

Hence, in recent years there has been a fair amount of research on non-linear noise removal techniques and prominent among them are the wavelet based denoising techniques. The disadvantage of Wavelet transform is many wavelet coefficients are needed to account for edges i.e. singularities along lines or curves which results into relatively high mean squared error (MSE). A new approach to image denoising is based on a recently introduced family of transforms e.g. curvelet transform which have been proposed as alternatives to wavelet representation of image data. Unlike wavelet, curvelet transforms accurately represent smooth functions using only a few nonzero coefficients.

The aim of this paper is to analyze the importance of the newly developed multiscale representation system, namely, the curvelet transform in terms of its two digital implementations i.e. transformation based on unequally-spaced fast fourier transforms (USFFT) and transformation based on wrapping of specially selected fourier samples and to find the best one for denoising a wide variety of gray scale images.

The curvelet transform was developed in last few years in an attempt to overcome inherent limitations of traditional multiscale representations such as wavelets. The curvelet transform is a multiscale pyramid with many directions and positions at each length scale, and needle-shaped element at fine scales. Curvelets have useful geometric features that set them apart from wavelets and the likes. For instance, curvelets obey a parabolic scaling relation which says that at scale 2^{-j}, each element has an envelope which is aligned along a 'ridge' of length $2^{-j/2}$ and width 2^{-j}.

II. Image Denoising

Image denoising can be formally defined as removal of noise present in the image while preserving the important and sharp features of the image. In acquiring, transmitting or processing a digital image for example, the noise induced degradation may be dependent or independent of data which is shown in fig. 1, where noisy image includes the original image and independent identically distributed noise process (n) with variance σ^2.

Priti Naik is an M.E. (Computer Engineering) student of Thadomal Shahani Engineering College, Mumbai, Maharashtra, India (e-mail: pritidipaknaik@gmail.com).

Shalini Bhatia is the Head, Computer Engineering Department, Thadomal Shahani Engineering College, Mumbai, Maharashtra, India (e-mail: shalini.tsec@gmail.com).

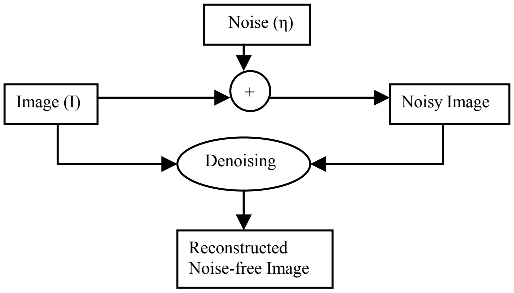


Fig. 1. Block diagram of Image Denoising Process

The goal of image denoising is to find an estimate of noise-free image based on the knowledge of noise [7]. A more precise explanation of the curvelet based denoising procedure can be given as follows. Curvelet transform is applied to a noisy image. The image I has an image function u(x,y) as a union of modified copies of itself. The net result is that target u is approximated by the attractive fixed point of curvelet transform T that performs the thresholding operation on the image function.

III. CURVELET TRANSFORM

The new ridgelet [5] and curvelet transforms [1] [2] [3] [4] were developed over several years in an attempt to break an inherent limit plaguing wavelet denoising of images. This limit arises from the well-known and frequently depicted fact that the two-dimensional (2-D) wavelet transform of images exhibits large wavelet coefficients even at fine scales, all along the important edges in the image, so that in a map of the large wavelet coefficients one sees the edges of the images repeated at scale after scale. While this effect is visually interesting, it means that many wavelet coefficients are required in order to reconstruct the edges in an image properly. With so many coefficients to estimate, denoising faces certain difficulties. There is, owing to well-known statistical principles, an imposing tradeoff between parsimony and accuracy which even in the best balancing leads to a relatively high mean squared error (MSE). While this tradeoff is intrinsic to wavelet methods (and also to Fourier and many other standard methods), there exist, on theoretical grounds, better denoising schemes for recovering images which are smooth away from edges. For example, asymptotic arguments show that, in a certain continuum model of treating noisy images with formal noise parameter ϵ, for recovering an image which is C^2 smooth away from edges, the ideal MSE scales like $\epsilon^{4/3}$ whereas the MSE achievable by wavelet methods scales only like ϵ. To approach this ideal MSE, one should develop new expansions which accurately represent smooth functions using only a few nonzero coefficients. Since, so few coefficients are required either for the smooth parts or the edge parts, the balance between parsimony and accuracy will be much more favorable and a lower MSE results. The ridgelet transform and curvelet transform were developed explicitly to show that this combined sparsity in representation of both smooth functions and edges is possible. The continuous ridgelet transform provides a sparse representation of both smooth functions and of perfectly straight edges.

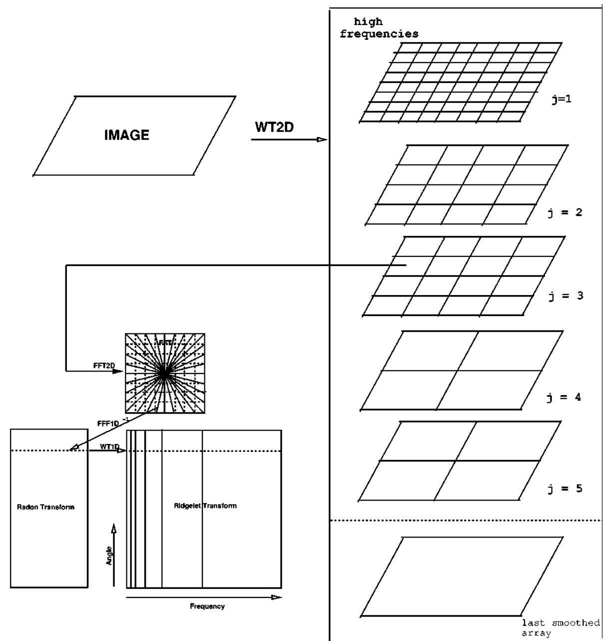


Fig. 2. Curvelet transform flowgraph

The original curvelet transform uses a preprocessing step involving a special partitioning of phase-space followed by the ridgelet transform, which is applied, to blocks of data that are well localized in space and frequency. The curvelet transform opens up the possibility to analyse an image with different block sizes, but with a single transform. The idea is to first decompose the image into a set of wavelet bands, and to analyze each band by a ridgelet transform. The block size can be changed at each scale level.

As shown in fig. 2 the curvelet transform is made up of a chain of steps. It uses à trous wavelet transform algorithm to decompose an n by n image I into J+1 subband arrays of size n x n. The figure illustrates the decomposition of the original image into subbands followed by the spatial partitioning of each subband. The ridgelet transform is then applied to each block.

The curvelet transform is mathematically valid and has a very promising potential in traditional application areas for wavelet-like ideas such as image processing, data analysis, and scientific computing. To realize this potential thoroughly and deploy this technology to a wide range of problems one would need a fast and accurate discrete curvelet transform operating on digital data. Original construction of curvelet transform is redundant and hence slow.

Two new fast digital implementations of a curvelet transform [4] i.e. transformation based on unequally-spaced fast fourier transforms (USFFT) and transformation based on wrapping of specially selected fourier samples are simpler, faster and less redundant than existing proposals and can be used for denoising. The two implementations essentially differ by the choice of spatial grid used to translate curvelets at each scale and angle. Both digital transformations return a table of digital curvelet coefficients indexed by a scale parameter, an orientation parameter and a spatial location parameter.

A. Digital Curvelet Transform via Unequispaced FFT's The digital coronization suggests Cartesian curvelets of the form

$$\psi_{j,l,k}(x) = 2^{3j/4}\psi_j\left(S_{\theta l}^{\ T}\left(x - S_{\theta l}^{\ -T} b\right)\right) \quad (1)$$

where b takes on the discrete values $b = (k_1 \cdot 2^{-j}, k_2 \cdot 2^{-j/2})$. The goal is to find a digital analog of the coefficients now given by

$$C(j,l,k) = \int \hat{} f\ (\omega)\tilde{}U_j\left(S_{\theta l}^{-1}\omega\right)e^{i\langle b, S_{\theta l}^{-1}\omega\rangle} d\omega \quad (2)$$

Suppose for simplicity that $\theta_l = 0$. To numerically evaluate above with discrete data, one would just (1) take the 2D FFT of the object f and obtain ˆf, (2) multiply ˆf with the window ˜U_j, and (3)take the inverse Fourier transform on the appropriate Cartesian grid $b = (k_1 \cdot 2^{-j}, k_2 \cdot 2^{-j/2})$. The difficulty here is that for $\theta_l \neq 0$, it is desired to evaluate the inverse discrete Fourier transform (DFT) on the nonstandard sheared grid $S^{-T}_{\theta l}\ (k_1 \cdot 2^{-j}, k_2 \cdot 2^{-j/2})$ and unfortunately, the classical FFT algorithm does not apply. To recover the convenient rectangular grid, however, one can pass the shearing operation to ˆf and rewrite the above equation as

$$C(j,l,k) = \int f(\omega) U_j\left(S_{\theta l}^{-1}\omega\right)e^{i\langle b, S_{\theta l}^{-1}\omega\rangle} d\omega = \int f\left(S_{\theta l}\ \omega\right) U_j(\omega) e^{i\langle b,\omega\rangle} d\omega \quad (3)$$

Suppose now that $f[t_1, t_2]$, $0 <= t_1, t_2 < n$ be the Cartesian array and let ˆ $f[n_1, n_2]$ denote its 2D discrete Fourier transform

$$f[n_1, n_2] = \sum_{t_1, t_2 = 0} f[t_1, t_2] e^{-t 2\pi (n_1 t_1 + n_2 t_2)/n} \quad (4)$$

Where $-n/2 \leq n_1, n_2 < n/2$

Which here and below, view as samples

$$f[n_1, n_2] = f(2\pi n_1, 2\pi n_2) \quad (5)$$

From the interpolating trigonometric polynomial, also denoted ˆ f, and defined by

$$\hat{} f(\omega_1, \omega_2) = \sum_{0 \leq t_1, t_2 < n} f[t_1, t_2] e^{-i(\omega_1 t_1 + \omega_2 t_2)/n} \quad (6)$$

the FDCT via USFFT simply evaluates

$$C^D(j,k,l) = \sum_{n_1, n_2 \in P_j} f[n_1, n_2 - n_1 \tan\theta_1] U_j[n_1, n_2] e^{i2\pi(k_1 n_1 / L_{1,j} + K_2 n_2 / L_{2,j})} \quad (7)$$

B. Digital Curvelet Transform via Wrapping

The 'wrapping' approach assumes the same digital coronization as in Section *A*, but makes a different, somewhat simpler choice of spatial grid to translate curvelets at each scale and angle.

The previous approach showed that it was possible to design curvelets with anisotropic spatial spacing of about $n/2^j$ in one direction and $n/2^{j/2}$ in the other, this approach would seem to require a naive regular rectangular grid with side-length about $n/2^j$ in both directions. In other words, one would need to compute on the order of 2^{2j} coefficients per scale and angle as opposed, to only about $2^{3j/2}$ in the USFFT-based implementation. By looking at fine scale curvelets such that $2^j \approx n$, this approach would require $O(n^{2.5})$ storage versus $O(n^2)$ for the USFFT version.

It is possible, however, to downsample the naive grid, and obtain for each scale and angle a subgrid which has the same cardinality as that in use in the USFFT implementation. The idea is to periodize the frequency samples.

As before, let $P_{j,l}$ be a parallelepiped containing the support of discrete-time localizing window ˜$U_{j,l}[n_1, n_2]$. Suppose that at each scale j, there exist two constants $L_{1,j} \sim 2^j$ and $L_{2,j} \sim 2^{j/2}$ such that, for every orientation θ_l, one can tile the two-dimensional plane with translates of $P_{j,l}$ by multiples of $L_{1,j}$ in the horizontal direction and $L_{2,j}$ in the vertical direction. Equipped with this notion, then define the *wrapping* W ˜ $U_{j,l}$ of the array ˜ $U_{j,l}$ around the origin by

$$W\tilde{}U_{j,l}\left[n_1 \bmod L_{1.j}, n_2 \bmod L_{2,j}\right] = \tilde{}\ U_{j,l}[n_1, n_2] \quad (8)$$

with $(n_1,n_2) \in P_{j,l}$. Hence, the wrapping transformation is a simple re-indexing of the array. Given $(n_1, n_2) \in P_{j,l}$, the correspondence between the wrapped and the original indices is one-to-one. For those angles in the range $\theta \in (-\pi/4, 3\pi/4)$, the wrapping is similar, after exchanging the role of the coordinate axes. The 2D inverse FFT of the wrapped array therefore reads

$$C^D(j,k,l) = 1/n^2 \sum_{n_1/2-1} \sum_{n_2/2-1} \tilde{}U_{j,l}[n_1, n_2] e^{i2\pi(k_1 n_1 / L_{1,j} + K_2 n_2 / L_{2,j})} \quad (9)$$

Where $n_1 = n/2, n_2 = -n/2$

IV. FILTERING

Here digital transforms for removing noise from image data are applied. If noisy data of the form

$$X_{i,j} = f(i,j) + \sigma Z_{i,j}, \qquad \text{is given}$$

Where f is the image to be recovered and z is the white noise, i.e. $z_{i,j} \sim N(0,1)$. There are two filtering methods used to denoise the noisy image i.e. Hard Thresholding and Partial Reconstruction.. Unlike FFT's or FWT's, discrete curvelet transform is not norm-preserving and, therefore, the variance of the noisy curvelet coefficients will depend on the curvelet index λ. Letting F denote the discrete curvelet transform matrix, we have $Fz \sim N(0, FF^T)$. Because the computation of FF^T is prohibitively expensive, an approximate value $\tilde{\sigma}^2_\lambda$, of the individual variances using norm of each individual curvelet was calculated.

Let y_λ be the noisy curvelet coefficients ($y = F_x$). Let's use the following hard-thresholding rule for estimating the unknown curvelet coefficients:

$$\tilde{Y}_\lambda = Y_\lambda \qquad \text{if } |Y_\lambda|/\sigma \geq K\tilde{\sigma}_\lambda \quad (10)$$

$$\tilde{Y}_\lambda = 0 \qquad \text{if } |Y_\lambda|/\sigma < K\tilde{\sigma}_\lambda \quad (11)$$

In this experiment, a scale-dependent value for k was chosen, where k = 4 for the first scale (j = 1) while k = 3 for the others (j > 1).

In partial reconstruction the image was reconstructed using few largest coefficients. And the remaining coefficients were set to zero. Important thing in this method is to select the percentage value of largest coefficients, which will be used for reconstruction of image. By trial and error the value selected is 6% and the threshold is then set.

Rule for estimating the unknown curvelet coefficients was used as follows:

$$\tilde{Y}_\lambda = Y_\lambda \quad \text{if} \quad |Y_\lambda| \geq T \qquad (12)$$

$$\tilde{Y}_\lambda = 0 \quad \text{if} \quad |Y_\lambda| < T \qquad (13)$$

V. RESULTS AND DISCUSSION

In this paper four methods have been implemented for image denoising. Two fast curvelet transform methods have been combined with two filtering methods.

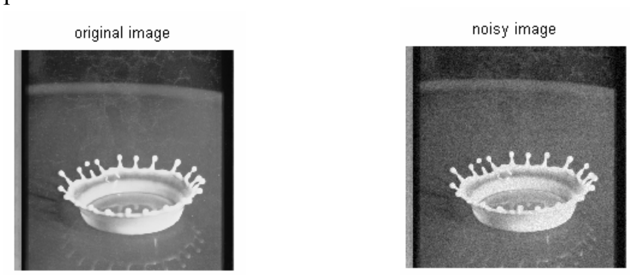
METHOD 1: Digital Curvelet Transform via Wrapping using Hard thresholding.

METHOD 2: Digital Curvelet Transform via Wrapping using Partial Reconstruction.

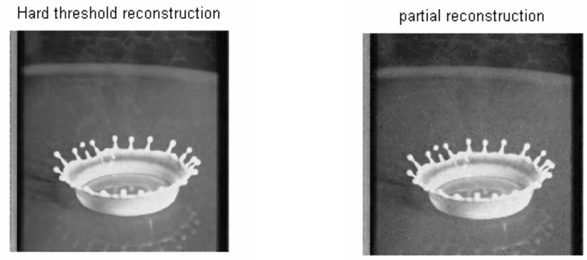
METHOD 3: Digital Curvelet Transform via USFFT using Hard thresholding.

METHOD 4: Digital Curvelet Transform via USFFT using Partial reconstruction

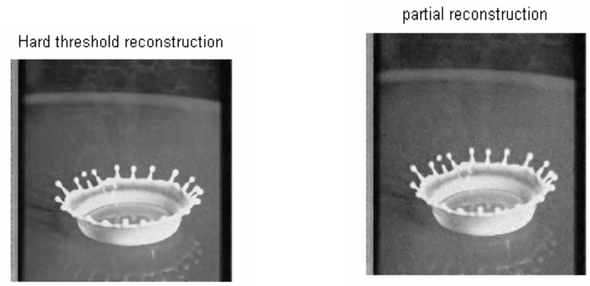
For this experiment 10 input images of 512 by 512 were used and 4 input images of 256 by 256 were used. Method 1 and method 3 gave better MSE and PSNR. This is because both the methods use same filtering method i.e. hard thresholding. In hard thresholding all coefficients of image obtained by curvelet transform are considered and respective thresholded values are used for reconstruction of image. As size of the image reduces, quality of image reconstructed becomes poor. Percentage reduction in MSE by each method is greater than that of σ = 10 but still less than the percentage reduction in MSE for 512 by 512 images. Table I and II shows MSE and PSNR obtained by each method for 512 by 512 images with standard deviation as 10. Results obtained by method 1 and method 3 are better than method 2 and 4. It has been observed that as curvelet transform with hard thresholding gives better results than curvelet transform with partial reconstruction.



(a) (b)



(c) (d)



(e) (f)

Fig. 3. Results obtained by four methods for noisy splash image with σ = 10 and image size 512 x 512. (a) Original splash (b) noisy splash with MSE = 98.76 and PSNR = 28.18 (c) Method 1 reconstruction with MSE = 22.74 and PSNR = 34.56 (d) Method 2 reconstruction with MSE = 41.18 and PSNR = 31.98 (e) Method 3 reconstruction with MSE = 23.07 and PSNR = 34.49 (f) Method 4 reconstruction with MSE = 43.34 and PSNR = 31.76

TABLE I
COMPARISON OF MSE IN NOISY IMAGE WITH FOUR METHODS FOR 512 X 512 IMAGES AND NOISE ADDED IN INPUT IMAGE (STANDARD DEVIATION = 10).

Image	MSE of Noisy Image σ = 10	MSE of Image reconstructed by			
		Method 1	Method 2	Method 3	Method 4
Sailboat	100.54	62.47	125.33	62.81	162.45
Splash	98.76	22.74	41.18	23.07	43.34
Lena	88.95	27.93	43.65	28.16	51.49
Elaine	100.66	44.41	60.07	44.84	64.90
Karishma	77.96	34.35	46.46	34.90	52.28
Barbara	88.44	77.29	98.97	77.93	120.34
Boat	100.16	56.88	84.18	57.08	102.96
OM	87.39	47.29	59.08	47.97	70.46
Pepper	99.32	37.46	53.17	38.04	62.12
Earth	100.00	37.68	74.41	38.46	92.48

TABLE II
COMPARISON OF PSNR IN NOISY IMAGE WITH FOUR METHODS FOR 512 X 512 IMAGES AND NOISE ADDED IN INPUT IMAGE (STANDARD DEVIATION = 10).

Image	PSNR of Noisy Image $\sigma = 10$	PSNR of Image reconstructed by			
		Method 1	Method 2	Method 3	Method 4
Sailboat	28.10	30.17	27.15	30.15	26.02
Splash	28.18	34.56	31.98	34.49	31.76
Lena	28.63	33.66	31.73	33.63	31.01
Elaine	28.10	31.65	30.34	31.61	30.00
Karishma	29.21	32.77	31.46	32.70	30.94
Barbara	28.66	29.24	28.17	29.21	27.32
Boat	28.12	30.58	28.87	30.56	28.00
OM	28.71	31.38	30.41	31.32	29.65
Pepper	28.16	32.39	30.87	32.32	30.19
Earth	28.13	32.36	29.41	32.28	28.47

VI. CONCLUSION AND FUTURE WORK

Although both the transforms have low running times, the USFFT transform is somewhat slower; this is due to the interpolation step in forward transform and to the CG iteration in the inverse transform. Hence the conclusion is that for any image size, curvelet transform based on wrapping using hard thresholding provides faster and better way to denoise the noisy image.

In future image quality of reconstructed images obtained by these strategies can be improved by varying filtering methods. Noise other than Gaussian white noise e.g. salt and pepper noise can also be introduced in the images used for denoising. These denoising techniques can be applied to color images also.

VII. REFERENCES

[1] E. Candes, and D. L. Donoho, “Curvelets-surprisingly effective nonadaptive representation for objects with edges”, in curve and surface fitting: SAINT Malo 1999, A. Cohen, C. Rabut, and L. L. Schumaker, Eds. Nashville, TN: Vandrbilt Univ. press 1999.

[2] J. L. Starck, E. Candes, and D. L. Donoho, “The Curvelet Transform for Image Denoising”, IEEE Transaction on Image Processing,Vol. 11,No. 6,June 2002.

[3] D. L. Donoho and Mark Duncan, “Digital Curvelet Transform: Strategy, implementation and experiments”, Stanford university November 1999.

[4] E. Candes, and D. L. Donoho, “Fast discrete Curvelet Transforms”, Stanford University, July 2005

[5] Philippe Carr’e and David Helbert,”Ridgelet Decomposition: Disceret implementation and color denoising”, September 2003

[6] Lakhvindar Kaur, Savita Gupta and R.C. Chauhan, “Image denoising using Wavelet Thresholding”

[7] K.P Soman and K.I Ramchandran.,“Insight into Wavelets from theory to practice”

[8] Anil K. Jain “Fundamental of digital Image Processing” University of California, 1997

[9] Gonzalez and Woods “Digital Image Processing” edition, 1997

[10] J. L. Starck, E. Candes, and D. L. Donoho, “The Astronomical Image Representation by the Curvelet Transform”, Astronomy and Astrophysics, in press, November 5, 2002.

VIII. BIOGRAPHIES

Shalini Bhatia was born on August 08, 1971. She received the B.E. degree in Computer Engineering from Sri Sant Gajanan Maharaj College of Engineering, Amravati University, Shegaon, Maharashtra, India in 1993, M.E. degree in Computer Engineering from Thadomal Shahani Engineering College, Mumbai, Maharashtra, India in 2003.

She has been associated with Thadomal Shahani Engineering College since 1995, where she has worked as Lecturer in Computer Engineering Department from Jan 1995 to Dec 2004 and as Assistant Professor from Dec 2004 to Dec 2005. Since Jan 2006 she is looking after the department as the Head. Her research interests include neural networks, fuzzy systems, bioinformatics, intelligent systems, distributed computing, image processing, and advanced computer architecture. She has published a number of technical papers in National and International Conferences. She is an active member of CSI and also a member of Special Interest Group in Artificial Intelligenge (SIGAI) which is a part of CSI**.**

Priti Naik was born on July 12, 1981. She received the B.E. degree in Computer Engineering from Thadomal Shahani Engineering College, Mumbai University, Bandra (W), Maharashtra, India in 2002. She is pursuing M.E. degree in Computer Engineering from Thadomal Shahani Engineering College, Mumbai, Maharashtra, India.

She has been working with Patni Computer Systems, Mumbai, since 2005, where she has worked as programmer, developer and mentor in mainframe technology. Currently she is performing the role of Group leader. She has worked with Bhausaheb Vartak Polytechnic, Vasai, Mumbai, from 2003 to 2005 as Lecturer in Computer Engineering Department. Her research interests include Computer Networks, Image Processing, and Database Management Systems.