Curvelet Based Multiresolution Analysis of Graph Neural Networks

Bharat Bhosale[1*]
[1]S. H. Kelkar College of Arts, Commerce and Science, University of Mumbai, Devgad 416613 (M.S.), India.

* Corresponding author. email: bn.bhosale@rediffmail.com
Manuscript submitted June 9, 2014; accepted September 10, 2014.
doi: 10.7763/IJAPM.2014.V4.304

Abstract: Multiresolution techniques are deeply related to image/signal processing, biological and computer vision, scientific computing, optical data analysis. Improving quality of noisy signals/images has been an active area of research in many years. Although wavelets have been widely used in signal processing, they have limitations with orientation selectivity and hence, they fail to represent changing geometric features along edges effectively. Curvelet transform on the contrary exhibits good reconstruction of the edge data by incorporating a directional component to the conventional wavelet transform and can be robustly used in the analysis of complex neural networks; which in turn are represented by graphs, called Graph Neural Networks.

This paper explores the application of curvelet transform in the analysis of such complex networks. Especially, a technique of Fast Discrete Curvelet Transform de-noising with the Independent Component Analysis (ICA) for the separation of noisy signals is discussed. Two different approaches viz. separating noisy mixed signals using fast ICA algorithm and then applying Curvelet thresholding to de-noise the resulting signal, and the other one that uses Curvelet thresholding to de-noise the mixed signals and then the fast ICA algorithm to separate the de-noised signals are presented for the purpose. The Signal-to-Noise Ratio and Root Mean Square Error are used as metrics to evaluate the quality of the separated signals.

Key words: Curvelet transform, graph neural networks, curvelet thresholding, denoising.

1. Introduction

Many underlying relationships among data in several areas of science and engineering, e.g. computer vision, molecular chemistry, molecular biology, pattern recognition, data mining, can be represented in terms of graphs. The complex networks like neural networks are also represented with the help of graphs by showing the computational elements, neurons of the network. Network analysis has many practical applications, for example, to model and analyze traffic networks. The applications include analysis to determine structural properties of a network such as the distribution of vertex degrees and the diameter of the graph, to find a measurable quantity within the network, for example, for a transportation network, the level of vehicular flow within any portion of it, and the analysis of dynamical properties of networks.

In recent years many important properties of complex networks have been delineated and studied the relationship between the structural properties, nature of dynamics taking place on these networks. For instance, the 'synchronizability' of complex networks of coupled oscillators can be determined by graph spectral analysis. These developments in the theory of complex networks have inspired new applications in the field of neuroscience such as study of models of neural networks, anatomical connectivity, and functional connectivity based upon functional magnetic resonance imaging (fMRI), electroencephalography (EEG) and magnetoencephalography (MEG). The recent applications of network theory to neuroscience such as modeling of neural dynamics on complex networks; graph theoretical analysis of neuroanatomical networks; and applications of graph analysis to study functional connectivity with fMRI, EEG and MEG are discussed at length [1].

A graph is an abstract representation of complex network. Many types of relations and process dynamics in physical, biological, social and information systems can be modeled with graphs. Graph analysis has been used in the study of models of neural networks, anatomical connectivity, and functional connectivity. These developments in the theory of complex networks have inspired new applications in the upcoming field of neuroscience including neural networks. Many practical problems can be represented by graphs that can be used to model different types of relations and process dynamics in physical, social and information systems. In mathematics, graphs are useful in geometry and certain parts of topology, e.g. Knot Theory. Algebraic graph theory has close links with group theory. In computer science, graphs are used to represent networks of communication, data organization, computational devices, the flow of computation, etc. Graph theory is also used to study molecules in chemistry and physics. In condensed matter physics, the three dimensional structure of complicated simulated atomic structures can be studied quantitatively by gathering statistics on graph-theoretic properties related to the topology of the atoms. For example, Franzblau's shortest-path (SP) rings. In chemistry, a graph makes a natural model for a molecule, where vertices represent atoms and edges bonds. This approach is especially used in computer processing of molecular structures, ranging from chemical editors to database searching. In statistical physics, graphs can represent local connections between interacting parts of a system, as well as the dynamics of a physical process on such systems. Likewise, graph theory is useful in biology and conservation efforts where a vertex can represent regions where certain species exist (or habitats) and the edges represent migration paths, or movement between the regions. This information is important when looking at breeding patterns or tracking the spread of disease, parasites or how changes to the movement can affect other species. In sum, many underlying relationships among data in several areas of science and engineering, e.g., computer vision, molecular chemistry, molecular biology, pattern recognition, and data mining, can be represented in terms of graphs and the graph theoretic approach can be employed to analyse such complex data.

In recent years many important properties of complex networks have been delineated and studied the relationship between the structural properties, nature of dynamics taking place on these networks. For instance, the 'synchronizability' of complex networks of coupled oscillators can be determined by graph spectral analysis. These developments in the theory of complex networks have inspired new applications in the field of neuroscience such as study of models of neural networks, anatomical connectivity, and functional connectivity based upon functional magnetic resonance imaging (fMRI), electroencephalography (EEG) and magnetoencephalography (MEG). The recent applications of network theory to neuroscience such as modelling of neural dynamics on complex networks; graph theoretical analysis of neuroanatomical networks; and applications of graph analysis to studies of functional connectivity with fMRI, EEG and MEG are discussed at length. The complex networks like neural networks are also represented with the help of graphs by showing the computational elements, neurons of the network. Each node corresponds to one neuron and the arrows usually denote weighted sums of the values from other neurons.

Recently various approaches have been unified in neural network model called graph neural networks (GNN), which is used for processing the data represented in graph domains. The GNN are of two kinds viz. Biological neural networks (BNN) and Artificial neural networks (ANN) [2]. BNNs are of objective existence, in which the neurons are linked as a network in a certain order, e.g. human neural network is the most intelligent network system. The ANN are aimed at modeling the organization principles of central neural system, with the hope that the biologically inspired computing capabilities of ANN will allow the cognitive and sensory tasks to be performed more easily and satisfactorily [3]. GNNs have been proved to be sort of universal approximator for functions on graphs and have been applied to several problems, including spam detection, object localization in images, molecule classification [4]. A novel neural network model that extends existing neural network methods for processing the data represented in the graph domain is considered in this work for curvelet based multi-resolution analysis.

Of late, wavelet based multi-resolution techniques are widely used in image/signal processing, biological and computer vision, scientific computing, optical data analysis. In our earlier works, we have also explored the applications of wavelet techniques in analyzing solitons [5], random processes [6], bio-informatics [7], neural networks [8]. Since Olshausen and Field's work in *Nature* [9], researchers in biological vision have discussed the similarity between vision and multi-scale image processing. However, wavelets do not provide a good direction selectivity, which is also an important response property of simple cells and neurons at stages of the visual pathway. To overcome this discrepancy, an anisotropic geometric wavelet transform, named ridgelet transform, was proposed by Candès and Donoho in1999 [10]. The ridgelet transform is optimal at representing straight-line singularities but global straight-line singularities are rarely observed in real applications. To analyze local line or curve singularities, the obvious way is to consider a partition of the image, and then to apply the ridgelet transform to the resulting sub-images. This block ridgelet-based transform, named curvelet transform, was first proposed by Candès and Donoho in 2000 [11]. Curvelet transform is a new extension of wavelet transform which aims to deal with interesting phenomena occurring along curved edges in 2D images. In the 2D case, the curvelet transform allows optimal sparse representation of objects with singularities along smooth curves. Moreover, the curvelet methods preserve the edges and the structures better than wavelet transform. Recently, denoising images using curvelet transform approach has been widely used in many fields for its ability to obtain high quality images [12]. Curvelet transform is a special multi-scale pyramid with many directions and positions at each decomposition scale. Therefore, the curvelet transform is more suitable than all other multi-scale transforms including wavelet in signal and image applications including, filtering, enhancement, compression, de-noising, and watermarking. The curvelets act as analytic signal filters in signal processing. The analytic signal filters are important in signal processing, because they yield a meaningful decomposition of filtered signal into amplitude and phase, where the amplitude has the interpretation of the envelope of the signal. We consider various forms of the curvelet transform in the following section.

2. Notations and Terminologies

In the most common sense of the term, a graph is an ordered pair $G = (E, V)$ comprising a set V of vertices or nodes together with a set E of edges or lines, which are 2-element subsets of V (i.e., an edge is related with two vertices, and the relation is represented as unordered pair of the vertices with respect to the particular edge). A vertex may exist in a graph and not belong to an edge. The presence of an edge between two vertices indicates the presence of some kind of interaction or connection between the vertices (the interpretation depends upon what is being modelled with the graph). The order of a graph $|V|$, is the number of vertices. A graph's size is $|E|$ the number of edges. The degree of a vertex is the number of edges that connect to it, where an edge that connects to the vertex at both ends (a loop) is counted twice.

As an extension of the simple graph is a weighted graph $G = (E, V, w)$ that consists of a set of vertices, a set of edges E, and a weight function

$w : E \rightarrow R^+$ which assigns a positive weight to each edge. The adjacency matrix A for a weighted graph G is the $N \times N$ matrix that contains the information about the connectivity structure of the graph. When an edge exists between two vertices i and j, the corresponding entry of the adjacency matrix is $A_{ij} = 1$; otherwise $A_{ij} = 0$. The likelihood $P(k)$ that a randomly chosen vertex will have degree k is given by the degree distribution: it is a plot of $P(k)$ as a function of k. The degree distribution can have different forms: Gaussian, binomial, Poisson, exponential or power law.

For a weighted graph, the degree of each vertex m, written as $d(m)$, is defined as the sum of the weights of all the edges incident to it. This implies $d(m) = \sum_n A_{m,n}$.

Every real valued function $f : V \rightarrow R$ on the vertices of the graph G can be viewed as a vector in R^N, where the value of f on each vertex defines each coordinate. This implies an implicit numbering of the vertices.

3. Curvelet Transform

Continuous Curvelet Transform: A Continuous Curvelet Transform (CCT) $f \rightarrow \Gamma_f(a, b, \theta)$ of functions $f(x_1, x_2)$ on R^2, into a transform domain with continuous scale $a > 0$, location $b \in R^2$, and orientation $\theta \in [0, 2\pi)$ is devised as follows.

Basically, the construction of curvelets involves two main ideas viz. one, considering polar coordinates in frequency domain and two, constructing curvelet elements being locally supported near wedges

As we work in R^2, we denote with $x = (x_1; x_2)^T$ the spatial variable, and $\xi = (\xi_1, \xi_2)^T$ the variable in frequency domain. The polar coordinates in frequency domain can be defined as $r = \sqrt{\xi_1^2 + \xi_2^2}$, $\omega = arctan\frac{\xi_1}{\xi_2}$.

Starting with a pair of windows $W(r)$ and $V(t)$, called as 'radial window' and 'angular window', respectively that obey the admissibility conditions

$$\int_0^\infty W(ar)^2 \frac{da}{a} = 1, \forall r > 0, \int_{-1}^1 V(t)^2 dt = 1 \tag{1}$$

These windows are both smooth, nonnegative and real-valued, with W taking positive real arguments and supported on $r \in (1/2, 2)$ and V taking real arguments and supported for

$$t \in [-1, 1].$$

For this purpose, the scaled Meyer window functions are considered the most fundamental window functions.

$$V(t) = \begin{cases} 1, |t| \leq 1/3 \\ \cos[\pi/2v(3|t| - 1)], 1/3 \leq |t| \leq 2/3 \\ 0, else \end{cases}$$

$$W(r) = \begin{cases} \cos[\pi/2v(5 - 6r)], 2/3 \leq r \leq 5/6 \\ 1,\ 5/6 \leq r \leq 4/3 \\ \cos[\pi/2v(3r - 4)], 4/3 \leq r \leq 5/3, 0, else \end{cases} \tag{2}$$

where v is a smooth function satisfying

$$v(z) = \begin{cases} 0, z \leq 0 \\ 1, z \geq 1 \end{cases},\ v(z) + v(1 - z) = 1, z \in R$$

The windows (2) will be used to construct a family of complex-valued waveforms with three parameters, the scale $a \in (0;\ 1]$, the location $b \in R^2$, and the orientation $\theta \in [0;\ 2\pi)$

Using the polar coordinates $(r;\ \omega)$ in frequency domain, the a-scaled window can be given as

$$U_a = a^{3/4} W(ar) V\left(\frac{\omega}{\sqrt{a}}\right), \tag{3}$$

for some a with $0 < a \leq 1$

This window U_a is applied for building curvelet functions as follows.

Let the basic element $\gamma_{a,0,0} \in L^2(R^2)$ be given by its Fourier transform $\hat{\gamma}_{a,0,0} = U_a(\xi)$ and let the curvelet family be generated by translation and rotation of the basic element $\gamma_{a,0,0}$,

$$\gamma_{a,b,\theta}(z) = \gamma_{a,0,0}\big(R_\theta(z-b)\big) \tag{4}$$

with the translation $b \in R^2$,

where $R_\theta = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$ is the 2×2 rotation matrix with angle θ.

Now use these windows in the frequency domain to construct a family of analyzing elements with three parameters: scale $a > 0$, location $b \in \boldsymbol{R^2}$ and orientation $\theta \in [0, 2\pi)$.

At scale a, the family is generated by translation and rotation of a basic element γ_{a00}:

$$\gamma_{ab\theta}(x) = \gamma_{a00}(R_\theta(x - b)) \tag{5}$$

where R_θ is the 2×2 rotation matrix effecting planar rotation by θ radians.

Equipped with this family of curvelets, define a CCT, Γ_f, a function on scale/location/direction space given by [6]

$$\Gamma_f(a,b,\theta) = \langle f, \gamma_{ab\theta}\rangle, a < a_0, b \in \boldsymbol{R^2}, \theta \in [0, 2\pi) \tag{6}$$

where $\langle f, \gamma_{ab\theta}\rangle$ is the inner products of function f onto analyzing elements, $\gamma_{ab\theta}$, called curvelets.

In other words, applying this family of high frequency elements

$\{\gamma_{a,b,\theta}\colon (0,1], b \in R^2, \theta \in [0,2\pi\}$, the CCT Γ_f of $f \in L^2(R^2)$, the curvelet transform is given as $\Gamma_f(a,b,\theta) = \langle \gamma_{a,b,\theta}, f\rangle = \int_{R^2} \gamma_{a,b,\theta}(x)\overline{f(x)}dx$.

The reproducible formula can also be given as

$$f(x) = \int \Gamma_f(a,b,\theta)\, \gamma_{ab\theta}(x) \mu(da\, db\, d\theta) \tag{7}$$

$$\|f\|_{L^2}^2 = \int \left|\Gamma_f(a,b,\theta)\right|^2 \mu(da\, db\, d\theta)$$

where, μ denotes the reference measure

$$d\mu = \frac{da}{a^3} db\, d\theta$$

This formula is valid for $f \in L^2$ that has a Fourier transform vanishing for $|\xi| < 2/a_0$.

A polar "wedge" supported by the radial window $\{W(r)\}$ and angular window $\{V(r)\}$, U_j in the Fourier domain is defined as

$$U_j(r,\omega) = 2^{-3j/4}\, W(2^{-j} r) V\left(\frac{2^{j/2}\omega}{2\pi}\right) \tag{8}$$

The basic curvelet is then defined as $\hat{\gamma}_{j,0,0}(\xi) = U_j(\xi)$, and the family of curvelet functions is given as

$$\gamma_{j,k,l}(x) = \gamma_{j,0,0}\left(R_{\theta_{j,l}}(x - b_k^{j,l})\right) \tag{9}$$

Further, to construct the discrete curvelet transform (DCT), choose the scales $a_j = 2^{-j}, j \geq 0$; the equidistant sequence of rotation angles $\theta_{j,l}$,
$\theta_{j,l} = \frac{\pi l 2^{-\lfloor j/2 \rfloor}}{2}$ with $l = 0,1,\ldots,4.2^{\lfloor j/2 \rfloor} - 1$; the positions $b_k^{j,l} = b_{k_1,k_2}^{j,l} = R_{\theta_{j,l}}^{-1}\left(\frac{k_1}{2^j}, \frac{k_2}{2^{j/2}}\right)$, with $k_1, k_2 \in Z$, and where R_θ denotes the rotation matrix with angle θ.

This choice will lead to a discrete curvelet system that forms a tight frame, i.e., every function $f \in L^2(R^2)$ will be representable by a curvelet series, $f = \sum_{j,k,l} \langle f, \gamma_{j,k,l} \rangle \gamma_{j,k,l}$, for which the Parseval identity $\sum_{j,k,l} |\langle f, \gamma_{j,k,l} \rangle|^2 = \|f\|_{L^2(R^2)}^2, \forall f \in L^2(R^2)$ holds. This ensures that the discrete curvelet transform will be invertible.

The terms in the above series, $c_{j,k,l}(f) = \langle f, \gamma_{j,k,l} \rangle$ are the curvelet coefficients. In particular, we can obtain $c_{j,k,l}(f)$ by Plancherel's Theorem for $j \geq 0$ as $c_{j,k,l}(f) = \int_{R^2} f(x) \overline{\gamma_{j,k,l}}(x) dx = \int_{R^2} \hat{f}(\xi) \overline{\hat{\gamma}_{j,k,l}}(\xi) d\xi = \int_{R^2} \hat{f}(\xi) U_j\left(R_{\theta_{j,l}} \xi\right) e^{i\langle b_k^{j,l}, \xi \rangle} d\xi$,
where

$$\hat{f}(\xi) = \frac{1}{2\pi} \int_{R^2} f(x) e^{-i\langle x, \xi \rangle} dx. \tag{10}$$

Yet another version of discrete curvelet transform, called Fast Discrete Curvelet Transform (FDCT), is based on wrapping of Fourier samples; which takes a 2D signal/image as an input in the form of a Cartesian array $f[m, n]$, where $0 \leq m < M$, $0 \leq n < N$ where M and N are the dimensions of the array. The outputs will be a collection of curvelet coefficients $c^D(j, l, k_1 k_2)$ indexed by a scale j, an orientation l and spatial location parameters k_1 and k_2.

$$c^D(j, l, k_1 k_2) = \sum_{0 \leq n \leq N}^{0 \leq m \leq M} f[m,n] \gamma_{j,k,k_1 k_2}^D [m,n] \tag{11}$$

Here, each $\gamma_{j,k,k_1 k_2}^D$ is a digital curvelet waveform, superscript D stands for "digital". Thus for any bivariate function, say, $f \in L^2(R^2)$ associated with signal vector, curvelet based sparse representation for the signal can be arrived at.

4. Graph Neural Networks

The neural networks can be represented by graphs showing the computational elements, neurons of the network, called as Graph neural network (GNN), which can be used to process structured data inputs, e.g. acyclic graph, cyclic graph, directed or un-directed graphs. This class of neural networks implements a function $\tau(G, n) \in R^m$ that maps a graph G and one of its nodes n onto an m-dimensional Euclidean space.

For a triplet, $G = (V; E; w)$ denoting the connected weighted graph (or a tree) with n nodes and m edges; wherein analogously, nodes being the neurons, arrows as the weighted sums of the values from other neurons, and $w = A_{ij}$, the weights showing a links between the nodes in the given neural network, a linear Gaussian factor analysis model that represents the neural network [13] can be given as,

$$x_i(t) = \sum_j A_{ij} s_j(t) + a_i + n_i(t) \tag{12}$$

where i indexes different components of the observation vector $x_{i,}$ representing weighted sums of underlying latent variables, j indexes different factors and A_{ij} are the weightings of the factors s, also known as factor loadings. The factors s and noise n are assumed to have zero mean. The bias in x is assumed to be caused by a. The effect of the inaccuracies and other causes is summarized by Gaussian noise n. In anticipation of the dynamic model, the observations are indexed by t referring to time, although in the usual factor analysis model, observations at different time instants are assumed to be independent of each other and the observations therefore need not form a sequence in time.

The corresponding vector representation can be given as

$$\mathbf{x}(t) = \mathbf{As}(t) + \mathbf{a} + \mathbf{n}(t) \tag{13}$$

where x, **s**, **a** and **n** are vectors and **A** is a matrix.

If the variances of the Gaussian noise terms $n_i(t)$ are denoted by σ_i^2, the probability which the model gives for the observation $x_i(t)$ can be written as

$$P(x_i(t) \mid s(t), \mathbf{A}, a_i, \sigma_i^2) = \frac{1}{\sqrt{2\pi\sigma_i^2}} exp\left(-\frac{\left[x_i(t) - \sum_j A_{ij} s_j(t) - a_i\right]^2}{2\sigma_i^2}\right) \tag{14}$$

Or equivalently, we can write, $\mathbf{x}(t) \sim N(\mathbf{As} + \mathbf{a}, \boldsymbol{\sigma}^2)$, where the vector σ^2contains the variances σ_i^2, the variances of the Gaussian noise terms $n_i(t)$ as σ_i^2, For mathematical convenience, the factors are assumed to have Gaussian distributions in the standard factor analysis model.

It is to be noted that the Gaussian model for the factors leads to inability to separate the underlying causes from each other and therefore needs to consider models where the factors are not restricted to be Gaussian, independent component analysis (ICA) models depending. In the ICA community, the factors have been traditionally called sources because they are not just some arbitrary combinations of the underlying causes, but ideally at least, the original independent causes. The practical techniques for estimating the sources are known also as blind source or signal separation.

As most of the natural signals are assumed to have additive random noise, which is modeled as Gaussian noise, denoising is the first step to be considered before the signal data is analyzed. Digital signals are invariably contaminated by noise. Noise arises due to imperfect instruments used in signal processing, problems with the data acquisition process, and interference which can degrade the data of interest. Also, noise can be introduced due to compression and transmission errors [14]. Improving quality of noisy images has been an active area of research in many years. It is seen that removing additive Gaussian noise by nonlinear methods such as Wavelet denoising and Curvelet denoising had better results than classic approaches [15]. The overall noise characteristics in a signal depend on factors like type of sensor, exposure time, pixel dimensions, ISO speed, and temperature. Blind source separation (BSS) [16] is the method of extracting underlying source signals from a set of observed signal mixtures with little or no information to the nature of these source signals. Independent component analysis is used for finding factors or components from multivariate statistical data and is one of the many solutions to the BSS problem [17]. The various ICA algorithms extract source signals based on the principle of information maximization, mutual information minimization, maximum likelihood estimation and maximizing non-Gaussianity. ICA is widely used in statistical signal processing, medical image processing, economic analysis and telecommunication applications [18].

We can draw a clear analogy between linear Gaussian factor analysis model that represents the neural network and ICA model and therefore robustly apply ICA algorithm for analyzing GNN.

The basic ICA model can be stated as,

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{n}(t) \tag{15}$$

where $\mathbf{x}(t)$ is an N dimensional vector of the observed signals at the discrete time instant t, $\mathbf{A}$ is an unknown mixing matrix, $\mathbf{s}(t)$ is original source signal of $M \times N$ $(M \leq N)$ and $\mathbf{n}(t)$ is the observed noise vector and M is number of sources.

The linear Gaussian factor analysis model in vector form (11) that represents the GNN can be viewed as ICA model given by vector Eq.(13) with replacing $\mathbf{A}\mathbf{s}(t) + \mathbf{a}$ by $\mathbf{A}\mathbf{s}(t)$.

The purpose of the ICA is to estimate $\mathbf{s}(t)$, which is the original source signal from $\mathbf{x}(t)$, which is the mixed signal, i. e. it is equivalent to estimating the matrix $\mathbf{A}$. Assuming that there is a matrix $\mathbf{W}$, which is the de-mixing matrix or separation inverse matrix of $\mathbf{A}$, then the original source signal is obtained by: $\mathbf{s}(t) = \mathbf{W}\mathbf{x}(t)$

The ICA algorithm assumes that the mixing matrix $\mathbf{A}$ must be of full column rank and all the independent components $\mathbf{s}(t)$, with the possible exception of one component, must be non-Gaussian. Further, the number of observed linear mixtures M must be at least as large as the number of independent components N $(M \geq N)$.

The following two approaches are discussed. The results can be compared with practical implementation of each of them.

Fast ICA Algorithm: The fast ICA is the most popular algorithm used in various applications as it is simple, fast convergent, and computationally less complex. It is a fixed point iterative algorithm that uses a nonlinear function $g(y) = tanh\,(ay)$, which is applied to the separation vector $\mathbf{W}$ that is recalculated at each iteration of the algorithm. The fixed point algorithm is to iterate to obtain a global minimum. Once you determine the vector $\mathbf{W}$, it is pointing to one of the independent components. The steps to implement the fast ICA algorithm are as follows [19]:

1) Center the data to make its mean zero.
2) Whiten the data to give z.
3) Choose an initial (e.g., random) vector $\mathbf{w}$ of unit norm.
4) Let $\boldsymbol{w}^{+} = E\{\,\boldsymbol{z}g(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{z})\,\} - E\{\,g'(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{z})\,\}\boldsymbol{w}$.
5) Let $\boldsymbol{w} = \boldsymbol{w}^{+}/\,\boldsymbol{w}^{+}$.
6) If not converged, go back to step 4.

Curvelet Thresholding Approach: The curvelets are analytic signal filters. The analytic signal filters are important in signal processing, because they yield a meaningful decomposition of filtered signal into amplitude and phase, where the amplitude has the interpretation of the envelope of the signal. Digital signals are invariably contaminated by noise. Curvelet de-noising is a simple operation, which aims at reducing noise in a noisy image. It is performed by selecting the FDCT coefficients below a certain threshold and setting them to zero as follows:

$$y_\lambda = \begin{cases} y_\lambda, & |y_\lambda| \geq t_\lambda \\ 0, & |y_\lambda| < t_\lambda \end{cases}$$

where t_λ is the threshold and λ is the index.

The threshold used is $t_\lambda = k\sigma_\lambda\sigma$, for some scale k, where σ is an estimation of the standard deviation of the noise, and σ_λ is an approximation value for the standard deviation of each curvelet coefficient estimated by using the Monte-Carlo simulation .

Thresholding neural network involves step by step procedure: Noisy signal (Input)→Applying FDCT→ Noisy curvelet coefficients→Tuning threshold value using threshold neural network→De-noised curvelet coefficients→Inverting FDCT→De-noised signal (Output).

In curvelet de-noising, the noise standard deviation is estimated from the finest scale coefficients corresponding to the diagonal orientation using the Median Absolute Deviation (MAD) estimate, which is given by: $\sigma = \frac{MAD}{0.6745}$

Then, we use the inverse curvelet transform to get the de-noising image.

The various performance factors are then calculated using the following equations.

$$\text{SNR} = 10\log_{10}\left(\frac{\sum_{i=1}^{N} x^2(i)}{\sum_{i=1}^{N}[x(i)-y(i)]^2}\right) \tag{16}$$

where $x(i)$ is the original source signal, $y(i)$ is the separated signal, i is the sample index and N is the number of samples of the signal.

$$\text{RMSE} = \frac{1}{N}\sqrt{\sum_{i=1}^{N}[x(i)-y(i)]^2} \tag{17}$$

In both the approaches, we select the signals ($\mathbf{x}(t)$in case of GNN) and add white Gaussian noise using a $(m \times n)$ random mixing matrix. Simulations can be performed on these noisy mixed signals so obtained using any Matlab® R 7.9 on a core i7 2.2 GHz PC.

In the first approach, noisy mixed signals can be obtained by applying the fast ICA algorithm first and then de-noising. The separated independent components can be thresholded using hard curvelet thresholding and the signals can be separated. The second approach uses hard curvelet thresholding to de-noise the signals, and then the fast ICA algorithm is used to separate the de-noised signals.

5. Conclusion

The curvelet transform is a multi-scale directional transform that allows an optimal non-adaptive sparse representation of objects with edges especially the higher dimensional signals. Major advantages of Curvelet are directionality and anisotropy. Although, wavelet allows us to analysis image in three different directions (Vertical, horizontal and Diagonal), the Curvelet supports more directions. Curvelet treatment is particularly effective at detecting signal/image activity along curves instead of radial directions, which are the most comprising objects like higher dimensional signals. It retains edge and detail information which is mostly lost in wavelet analysis. The Curvelet threshold technique devised in this work can be therefore robustly employed for analyzing complex neural networks, in particular and higher dimensional signals, in general, that arise in wide range of applications including image/video processing, seismic exploration, fluid mechanics, simulation of partial different equations, and compressed sensing.

Two different approaches of separation of noisy mixed signals have been discussed to observe the effect of de-noising before and after signal separation. De-noising was done using hard curvelet thresholding and separation was based in using ICA. In this work, thresholding neural network using Curvelet coefficients instead of wavelet coefficients is proposed. The results of the first approach (de-noising then separation) can be compared with that of the performance metrics (SNR and RMSE) of the second approach (separation then de-noising).

References

[1] Cornelis, S. J., & Jaap, R. C. (2007). Graph theoretical analysis of complex networks in the brain. *Nonlinear Biomedical Physics*, *1*, 3.

[2] Podolak, I. T. (1998). Functional graph model of a neural network. *IEEE Trans Syst Man Cybern B Cybern*, *28(6)*, 876-81.

[3] Xu, J., & Zheng, B. (2002). Neural networks and graph theory. *Science in China (Series F)*, *45(1)*, 1-24.

[4] Scarselli, F., Gori, M., *et al*. (2009). The graph neural network model, neural networks. *IEEE Transactions*, *20(1)*, 61-80.

[5] Bhosale, B., & Biswas, A. (2013). Multi-resolution analysis of wavelet like soliton solution of KdV equation. *International Journal of Applied Physics and Mathematics*, *3(4)*, 270-274.

[6] Bhosale, B. (2014). Wavelet analysis of randomized solitary wave solutions. *International Journal of Mathematical Analysis and Applications*, *1(1)*, 20-26.

[7] Bhosale, B., Ahmed, B. S., & Biswas, A. (2013). Wavelet based analysis in bio-informatics. *Life Sci*., *10(2)*, 853-859.

[8] Bhosale, B., Ahmed, B. S., & Biswas, A. (2013). On wavelet based modeling of neural networks using graph theoretic approach. *Life Sci*., *10(2)*, 1509-1515.

[9] Olshausen, B., & Field, D. (1996). Emergence of simple-cell receptive filed properties by learning a sparse code for natural images. *Nature*, *381*, 607–609.

[10] Candès, E., Demanet, L., Donoho, D., & Ying, L. (2006). Fast discrete curvelet transforms. *Multiscale Model Simul*., *5(3)*, 861–899.

[11] Candès, E., & Donoho, D. (2005). Continuous curvelet transform resolution of the wavefront set. *Appl. Comput. Harmon. Anal*., *19(2)*, 162–197.

[12] Kota, N., & Reddy, G. (2011). Fusion based gaussian noise removal in the images using curvelets and wavelets with gaussian filter. *International Journal of Image Processing*, *5(4)*, 230-238.

[13] Valpola, H. (2000). Bayesian ensembel learning for non-linear factor analysis. *Acta Ploytechnica*, *Scandinavica, Mathematics and Computing Series*, *108*, 54-64.

[14] Motwani, M., Gadiya, M., & Motwani, R. (2004). Survey of image denoising techniques. Proceedings of GSPx (pp. 27-30). CA.

[15] Yaser, N., & Mahdi, J. (2011). A novel curvelet thresholding function for additive gaussian noise removal. *International Journal of Computer Theory and Engineering*, *3(4)*, 169-178.

[16] Shehata, S., Diab, M., Salam, M., & Sayed M. (2013). Analysis of blind signal separation of mixed signals based on fast discrete curvelet transform. *International Electrical Engineering Journal*, *4(4)*, 1140-1146.

[17] Cardoso, J. F., & Souloumiac, A. (1993). Blind beam forming for non-gaussian signals. *IEEE Proceeding Part F*, *140(6)*, 362-370.

[18] Hyvärinen, A., & Oja, E. (2000). Independent component analysis: Algorithms and applications. *Neural Networks*, *13(4-5)*, 411-430

[19] Erikki, O., Aapo, H., & Juha, K. (2001). *Independent Component Analysis*. John Wiley &Sons, Inc.

Bharat Bhosale comes from Maharashtra, India. He earned his B.Sc (Hons), M.Sc, and Ph. D degrees in mathematics from Shivaji University, Kolhapur.

He has worked as an associate professor in mathematics in the University of Mumbai and at present he is working as the principal of S. H. Kelkar (undergraduate and post graduate) College, affiliated to University of Mumbai. He has visited several countries as an invited speaker at the international conferences / seminars. He has published 20 research papers in national / international journals and two books. He is a referee and reviewer of a number of international journals. His research areas are in integral transforms, wavelets, curvelets and its applications in signal/image processing. Currently he is working on wavelets, curvelets and solitons, neural networks.

Dr. Bhosale is a member of Indian Mathematical Society (IMA), International Association of Computer Science and Information Technology (IACSIT). He has been awarded with Bharat Gaurav citation for his contribution in the field of higher education.