Image De-noising using Spectral Graph Wavelet Transform

Abstract

Image denoising has always been one of the most important and trying problems in the field of image processing. In this experiment, yet another graph-based approach to denoise an Image is implemented. The major tool used is the Spectral Graph Wavelet Transform (SGWT).

Spectral Graph Wavelet Transform

A wavelet transform is another basis, which is used to represent any signal. The basis is a bunch of ‘wave-like’ functions, which are localized both in time and frequency. They are seen as different time-scaled and shifted functions of a single mother wavelet function.

Porting to Graphs:

Given an undirected, weighted graph, G, and its adjacency matrix, A, the graph signal can also be written in terms of different kinds of orthogonal bases – the graph Fourier transform being one such representation.

Similarly, the notion of wavelets can also be extended to graphs. Given a mother kernel, ‘g’ defined on all the vertices, the shifted and scaled functions make up the wavelet basis. Shifting can be interpreted in the vertex domain, but it’s a little difficult to define ‘scaling’ function on graphs. Thus, each scale of the kernel is looked at as another function to be operated on the graph. Thus, the SGWT coefficients can be computed as

$$W_f(t,n) = F^{-1} \left( \hat{g}(tl) \cdot \hat{f}(l) \right)$$

where ‘t’ is the scaling parameter, $\hat{f}(l)$ represents the graph fourier transform of f in the vertex domain and $l$ is the index given to the eigenvectors.

$$W_f(t,n) = F^{-1} \left( \hat{g}(tl) \cdot \hat{f}(l) \right) = g(tL)f = \sum_{l=0}^{N-1} g(t\lambda_l) \cdot \hat{f}(l) \cdot \chi_l(n)$$

where $\chi_l(n)$ are the eigenvectors indexed by $l$. 
Properties of SGWT:

1. For each scale $t$, there are $N$ coefficients and if there are $J$ scales, there are a total of $N*(J+1)$ coefficients. Thus, it is an over-complete transform unlike other methods.
2. SGWT coefficients are zero mean, centralized around each vertex and the kernel is chosen to be band-pass in the ‘scale’ (here, time domain)
3. Based in both vertex and time domains.
4. There is an inherent simplicity in constructing SGWT compared to other methods like the diffusion wavelets.

Approach

1. Take the clean version of the image, form an adjacency matrix based on the similarity between the patches non-locally, ie. $a_{ij}$,
   \[ a_{ij} = e^{-\|t(i)-t(j)\|/\sigma_h} * e^{-\|t(i)-j\|/\sigma_r} \]
   The graph is defined on this clean image, with adjacency matrix, $A$ as computed above.
2. The signal here is the noisy image. It is placed on the graph obtained in step 1.
3. SGWT is computed for the signal on the graph at various scales, chosen as 5 here.
4. Coefficients of SGWT are soft-thresholded with a cutoff value.
5. SGWT inverse is then computed on the above coefficients to obtain the denoised image.

Why should it work?

When the SGWT coefficients are computed based on the adjacency matrix from the clean image, everything except the noise is sparse or compressible. The noise elements stand out as the uncorrelated or the non-sparse entities. Thus, soft thresholding on the SGWT coefficients might help remove a portion of this. This was the main idea of the experiment.

Experiment

60x60 patches of the Lena image, both clean and noisy versions are used throughout.

After computing the adjacency matrix, $A$, the no. of neighbors is restricted to some number $M$, (20 here) to help with the computation and make the matrix sparser. $A$ is a 3600x3600 sparse matrix.

Chebyshev polynomial approximation of about order 25, was applied to the kernel to ensure a fast computation of the SGWT.

The SGWT toolbox provided by EPFL on their wiki site is used throughout in this experiment for sgwt related functionality.
Results

Clean Noisy Cleaned (th=60)

Cleaned (th=100) Cleaned (th=150)

Discussion

It can be seen that as the threshold, th, on the SGWT coefficients is increased, though the image starts becoming ‘warm’, like the ‘low-pass’ blurring that happens generally, it is visually more appealing than the nosy image.

Some of the challenges faced here are:

- The noise is still persistent to a great extent: simply thresholding isn’t enough. There needs to be some other way to manipulate these coefficients.
- Need apriori information of the clean image: maybe, in the next stage, the graph can be learned from various noisy images. Since across the images, it’s the image content that stays sparse, some kind of correlation between the images could help build the graph
- Constructing the adjacency matrix efficiently: local/non-local, hybrid/oriented local connectivity unions, etc can be chosen as explained in [2].
- Doesn’t apply to all kinds of noise distributions, only linear is what is expected.
References

2. Wavelets on Graphs via Spectral Graph Theory, D K Hammond, P Vanderheynst, R Gribonval
3. Image Denoising with non-local Spectral Graph Wavelets, David K Hammond, Laurent Jacques and Pierre Vandergheynst
   In "Image Processing and Analysis with Graphs : Theory and Practice", Edited by Olivier Lézoray and Leo Grady, CRC Press, 2012