MULTISCALE MANIFOLD REPRESENTATION AND MODELING

Hyeokho Choi and Richard Baraniuk

Rice University
Department of Electrical and Computer Engineering
6100 Main Street, Houston, Texas 77005, USA

ABSTRACT

Many real world data sets can be viewed as points in a higher-dimensional space that lie concentrated around a lower-dimensional manifold structure. We propose a new multiscale representation for such point clouds based on lifting and perfect matching. The result is an adaptive wavelet transform that decomposes a point cloud into manifold approximations and details at multiple scales. We illustrate with several examples that the transform can extract an unknown smooth manifold from noisy point cloud samples using simple wavelet thresholding ideas.

1. INTRODUCTION

Algorithms for processing high-dimensional data are becoming increasing important for applications such as computer vision, pattern recognition, learning theory, remote sensing, image processing, biomedicine, and computer networking. For example, in some cases an $n \times n$ image can be viewed profitably as a point in $\mathbb{R}^{n^2}$-dimensional space. However, even with today's most advanced computational tools, the dimensionality of the data is often prohibitively large for conventional multidimensional signal processing tools.

Handling huge high-dimensional data sets requires special algorithms that exploit the inherent structures hidden in the data. Due to the nature of the underlying physics that govern data generation, many practical data lie along a low-dimensional nonlinear *manifold* in the high-dimensional space.

In many cases of practical importance, we do not have an analytic formula describing the manifold but instead merely point samples that implicitly specifying the manifold. In many practical cases, the samples are noisy and thus form a fuzzy cloud of points around the unknown manifold. This motivates the problem of *manifold estimation from noisy samples*. Moreover, even when the samples are noise-free, the points can be highly redundant, which motivates the problem of *manifold representation and compression*.

In this paper, we take some first steps towards a *multiscale representation* for point clouds that are concentrated around piecewise smooth manifolds. Using a nonlinear lifting scheme [1] on the point locations, we develop a *wavelet transform* that adapts to the local manifold geometry. At each of a nested set of scales, the transform creates an increasingly smoothed, lower-resolution manifold (scaling coefficients) and a set of inter-scale differences (wavelet coefficients). For smooth and piecewise smooth manifolds, the wavelet coefficients are sparse, which allows us to concisely represent the manifold geometry. This sparsity can be exploited to extract the underlying manifold geometry from a noisy point cloud, effectively distinguishing the manifold geometry information from the noise. In short, the manifold wavelet transform allows us to apply the gamut of powerful wavelet-based signal processing tools on problems involving high-dimensional point clouds.

A related multiscale representation for point clouds [2] uses a simplified form of lifting to represent and then progressively encodes the point *locations*. The lifting algorithm proposed here is fundamentally different in that it provides a multiscale representation of the underlying manifold *geometry*, not the locations of the individual points.

This paper is organized as follows. Section 2 reviews lifting, and Section 3 develops our manifold wavelet transform. In Section 4, we apply the transform to extract a manifold from two noisy point clouds. Section 5 closes the paper with a discussion and directions for future research.

2. LIFTING

The lifting scheme [1] was developed to enable linear, nonlinear, and adaptive wavelet transforms on complex geometries and with nonuniformly sampled data. All of the above are very difficult to handle using traditional wavelet analysis based on filterbanks. As illustrated in Fig. 1(a), a typical lifting stage consists of three steps: Split, Predict, and Update. Let $x[n]$ be a signal.

Split: Divide $x[n]$ into its even and odd polyphase components $x_e[n]$ and $x_o[n]$, where $x_e[n] = x[2n]$ and $x_o[n] = x[2n+1]$.

Predict: Predict the odd samples $x_o[n]$ from the neighboring even samples $x_e[n]$. Denoting the predictor output by $P(x_e)[n]$, the prediction residual (the *wavelet coefficient*) is given by

$$d[n] = x_o[n] - P(x_e)[n]. \tag{1}$$

If the underlying signal is locally smooth, then the wavelet coefficient $d[n]$ will be small. The prediction procedure essentially applies a high-pass filter to $x[n]$.

Update: The final lifting step transforms the even samples $x_o[n]$ into a lowpass filtered and subsampled version of $x[n]$. By adding a linear combination of $d[n]$ to $x_e[n]$, we obtain the *scaling coefficients*

$$c[n] = x_e[n] + U(d)[n], \tag{2}$$

where $U(d)[n]$ is the output of the update block. After the update step, the wavelet and scaling coefficients are properly normalized.

Email: {choi, richb}@rice.edu. Web: dsp.rice.edu. This work was supported by NSF grant CCF-04-31150, ONR grant N00014-02-1–0353, AFOSR grant FA9550-04-10148, and the Texas Instruments Leadership University Program.

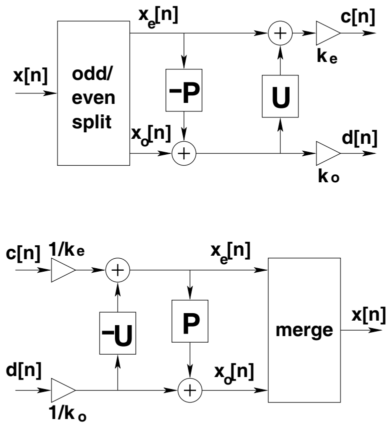

Fig. 1. *Lifting steps: Split, Predict, and Update. Inverse lifting steps: Undo update, Undo prediction, and Merge.*

To analyze the signal into multiple scales, we can repeatedly apply the lifting stages to the scaling coefficients. Each lifting step is easily invertible as we see in Fig. 1(b).

Any conventional wavelet transform can be implemented as a concatenation of lifting stages [1]. For example, the simplest Haar wavelet transform can be implemented as

$$d[n] = 0.5(x_o[n] - P(x_e)[n]) = 0.5(x_o[n] - x_e[n]) \quad (3)$$

and

$$c[n] = x_e[n] + U(d)[n] = x_e[n] + d[n]. \quad (4)$$

In our manifold wavelet transform, we will predict point samples using a local hyperplane fit to the local manifold geometry and an update similar to the Haar update in (4).

3. WAVELET TRANSFORM ON POINT SAMPLES

Using lifting, we now develop a multiscale representation for sampled d-dimensional manifolds in D-dimensional space ($d < D$). There are two important distinctions with the setting of conventional wavelet transforms. First, unlike signals and images that are *functions* defined over a fixed domain, here the manifold geometry is hidden in the *locations* of the point samples. Second, there is no natural organization of the sample points along the line or plane, complicating the definition of an "even" and "odd" split of the samples.

The Split step described in Sec. 2 pairs up neighboring even and odd coefficients. For smooth data, this results in the smallest possible wavelet coefficients. With point cloud data, how to perform such a pairing is not evident. Fortunately, the problem of optimally pairing an even number of points has received considerable attention in graph theory under the name *minimum-weight perfect matching* [3], and there are fast algorithms available [4].

Split: Our Split step pairs up an even number of points based on the Euclidean distances between points using perfect matching. The pairing is globally optimal in the sense that the sum of pairwise distances is minimum among all possible pairings. The pairing naturally splits the point set into two subsets of the same size. We randomly label the two points in each pair as even and odd. For example, in Fig. 2 the crosses and circles represent even and odd points, respectively. Note that each even point is paired with a nearby odd point. While the points do not have a natural ordering, we denote the paired even and odd points by $p_e[n]$ and $p_o[n]$ using an abstract index n.

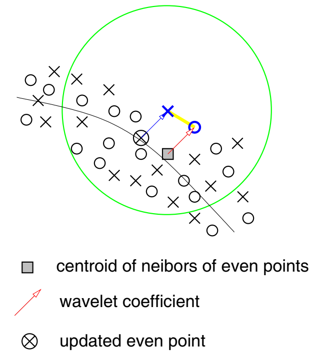

Fig. 2. *Lifting manifold samples. The points are split into even/odd pairs using perfect matching. Each odd point (blue circle) is predicted by computing the centroid (square) of the set comprising its even mate (blue cross) and its neighboring even points. The difference is the wavelet coefficient (red arrow). Then, each even point is updated by subtracting the wavelet coefficient to obtain the scaling coefficient (circle with cross).*

Predict: Once the split is achieved, we predict each odd point $p_o[n]$ (see Fig. 2) using nearby even points. Assuming that the points lie on a locally smooth manifold, it is natural to expect that the predictor should predict a point on the manifold that is close to the current odd point. We form a linear approximation to the local manifold geometry by fitting a d-dimensional hyperplane to the local subset of even points. The subset of even points can be easily obtained by considering the neighbors of the even point $p_e[n]$ that was paired up with $p_o[n]$ (all the even points in the green circle in Fig. 2). In our current implementation, we use all neighboring even points within a certain distance from $p_e[n]$. The radius of neighborhood should be set properly for different manifold structures and noise variance, because it governs the stability of the local plane fitting.

While any point on the hyperplane can be a candidate prediction, the simplest choice is the centroid of the neighboring even points. Note that the centroid lies on the least-squares best fit plane to the even points. When the sampling distribution is close to uniform, the centroid will be close to $p_o[n]$.

Given this predictor, the wavelet coefficient is defined as the difference between the current odd point and the predicted point. In summary, denoting the neighbors of $p_e[n]$ as $\mathcal{N}(p_e[n])$, we can write the prediction step as

$$d[n] = p_o[n] - \sum_{\mathcal{N}(p_e[n])} p_e[i]. \quad (5)$$

In Fig. 2, the red vector represents the wavelet coefficient $d[n]$. Note its short length.

Since the centroid of the neighboring even points is expected to be very close to the manifold, the wavelet coefficients thus encode the locations of the points *relative to the manifold*. Since the innovation in geometry introduced by each point is mainly contained in the direction orthogonal to the local manifold geometry, we can further decompose each wavelet coefficient $d[n]$ into normal and tangential components based on the local coordinates defined by the locally fit hyperplane. The *normal component* is the orthogonal projection of $d[n]$ onto the hyperplane's normal and is related to the manifold geometry. The *tangential component* is the orthogonal projection of $d[n]$ onto the hyperplane and encodes the distribution of the points on the manifold (with respect to the local centroids).

Update: The update step works on the even samples to create a smoothed and subsampled version of the original points for processing at the next coarse scale. For point samples on a manifold, smoothing means moving the points closer to the manifold. Recalling that the wavelet coefficients represent the offset of each odd point from the locally fit hyperplane and assuming that the current even point $p_e[n]$ to be updated is very close to the odd point $p_o[n]$ with which it has been paired, we can subtract the wavelet coefficient from the even point to move closer to the plane. This is similar to the update step corresponding to the Haar wavelet in (4) in that only a single wavelet coefficient is used to update. That is, the updated coarse scale points are given by (see Fig. 2)

$$c[n] = p_e[n] - d[n]. \quad (6)$$

Iterating on the scaling coefficients creates a multiscale transform with $2\times$ fewer points at each scale, independent of the dimensions d of the manifold and D of the ambient space. This is in sharp contrast to other multiscale techniques for multidimensional data that decimate by 2^D at each scale. To construct a complete binary tree of scaling and wavelet coefficients, we assume that the total number of samples is a power of two.

4. MANIFOLD EXTRACTION VIA WAVELET THRESHOLDING

The manifold wavelet transform provides a convenient framework for representing and processing manifold geometry at multiple scales. This can be useful for extracting and modeling different features of the manifold. In particular, since the wavelet coefficients represent prediction errors from a smoothed version of the manifold, thresholding the wavelet coefficients can remove noise in the sample locations, resulting in samples closer to the original manifold. In this section, we apply our transform to extract a smooth manifold from a noisy point cloud of samples.

4.1. Wavelet thresholding

The manifold wavelet transform provides a sparse approximation of the manifold geometry. When the points are noisy, the sparsity allows simple but effective noise removal by wavelet thresholding [5].

However, processing the wavelet coefficients of a manifold requires extra caution since each wavelet coefficient consists of two distinct parts: its tangential and normal components. Each wavelet coefficient is the prediction error resulting from predicting a point location from its neighbors. The prediction is possible because all of the points lie along a lower-dimensional manifold. However, while the points are close to the manifold, when projected onto the manifold, they will be distributed randomly on the manifold. This implies that only the normal components of wavelet coefficients are predictable. The tangential components are purely random and are related to the distribution of points along the manifold.

Thus, when we threshold the wavelet coefficients, we should retain the tangential components of wavelet coefficients in order to preserve the point distribution along the manifold. By thresholding the normal components, we draw the points closer to the manifold, effectively attenuating the noise in the point samples.

Figures 3 and 4 demonstrate denoising of manifold point samples through wavelet thresholding. The first two plots show the original smooth 2-dimensional manifold in $\mathbb{R}^3$ and 8192 points from the manifold with Gaussian noise added to their point locations. After computing 5 scales of lifting, we thresholded the normal components of the wavelet coefficients and inverted the transform to obtain the points shown in the last plot. Note that these points that are much closer to the original manifold.

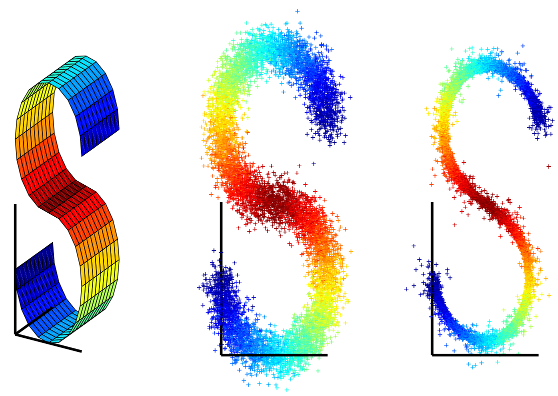

Fig. 3. *Thresholding result for a 2-dimensional surface in $\mathbb{R}^3$. Left: original manifold. Middle: noisy point samples. Right: points after wavelet thresholding. The two point sets are viewed along the y-axis to highlight their variability away from the underlying manifold.*

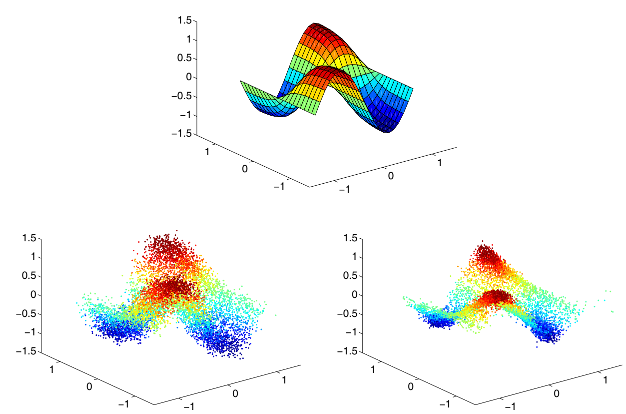

Fig. 4. *Thresholding result for a 2-dimensional surface in $\mathbb{R}^3$. Top: original manifold. Below left: noisy point samples. Below right: points after wavelet thresholding.*

4.2. Wedgelet manifold extraction

We now provide an example of *manifold extraction* from noisy samples in high-dimensions. Consider the manifold consisting of wedgelet image blocks [6]. A wedgelet is a square $n \times n$ block of image pixels containing an ideal step edge at a certain angle and offset from the center. Thus the collection of all wedgelets defines a locally 2-dimensional manifold in $\mathbb{R}^{n^2}$. In our experiment, we generated 5×5 pixel blocks of wedgelets with various edge orientations and offsets. The ideal edge is pixelized by computing the averaging an ideal wedgelet over each pixel.

First we generated 8192 5×5 ideal wedgelets having various orientations and offsets. Then, we added random Gaussian noise to each pixel to obtain the noisy wedgelets at the top of Fig. 5.

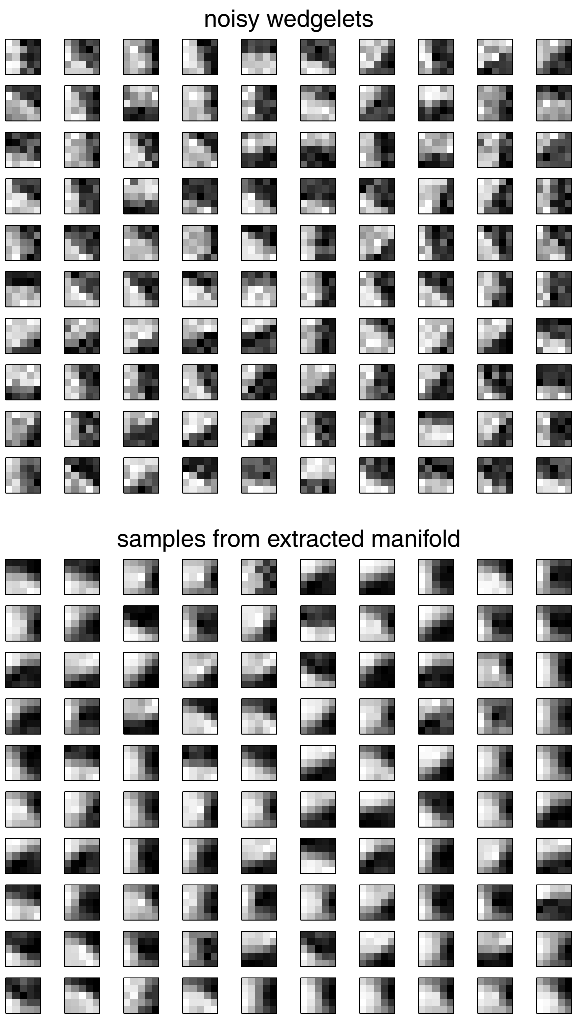

Fig. 5. Wedgelet manifold extraction from 8192 noisy 5×5 wedgelets of various orientations and offsets. Top: original noisy wedgelets (100 randomly picked). Bottom: Images corresponding to sample points after wavelet thresholding based manifold extraction (100 randomly picked).

After performing 5 stages of lifting, we thresholded the normal components of wavelet coefficients and inverted the lifting stages. The result is an estimate of the underlying manifold that exploits the fact that it is piecewise smooth. Note that the bottom images in Fig. 5 are *not* denoised versions of the top images. Instead, the bottom images are points that lie much closer to the underlying manifold than the points in the top images. Note that they are essentially smoothed wedgelets. This implies that the extracted manifold is a smoothed version of the ideal wedge manifold.

5. CONCLUSIONS

In this paper, we have developed a multiscale wavelet transform for point clouds lying along a nonlinear manifold in high-dimensional space. Using the lifting concept to adapt to the local manifold geometry, we obtain a multiscale representation of point samples from on manifold. Using simple wavelet thresholding, we were able to make a reasonable estimate of the manifold structure from a fuzzy cloud of points.

Although we obtained promising preliminary results, much work remains. Our proposed scheme is based on a linear approximation of the local manifold geometry. More accurate modeling using higher-order prediction and update steps would provide sparser representations. Furthermore, the asymptotic approximation rate requires further analysis. Finally, our current algorithm seems to be quite sensitive to the choice of neighbors in the prediction step. The optimal and adaptive choice of the neighbors to preserve stability based on the sample distribution in the point data set remains as an interesting future research topic.

6. REFERENCES

[1] W. Sweldens, "The lifting scheme: A custom-design construction of biorthogonal wavelets," *J. Appl. Comp. Harm. Anal.*, vol. 3, no. 2, pp. 186–200, 1996.

[2] M. Waschbüsch, M. Gross, F. Eberhard, E. Lamboray, and S. Würmlin, "Progressive compression of point-sampled models," in *Proceedings of the Eurographics Symposium on Point-Based Graphics*, 2004, pp. 95–102.

[3] J. Edmonds, "Paths, trees and flowers," *Candian Journal of Math*, vol. 17, no. 3, pp. 449–467, 1965.

[4] W. Cook and A. Rohe, "Computing minimum-weight perfect matching," *INFORMS Journal on Computing*, vol. 11, pp. 138–148, 1999.

[5] D. Donoho and I. Johnstone, "Ideal spatial adaptation via wavelet shrinkage," *Biometrika*, vol. 81, pp. 425–455, 1994.

[6] D. L. Donoho, "Wedgelets: Nearly-minimax estimation of edges," *Annals of Stat.*, vol. 27, pp. 859–897, 1999.