High-Dimensional Data Fusion via Joint Manifold Learning

Mark A. Davenport,[s] Chinmay Hegde,[r] Marco F. Duarte,[p] and Richard G. Baraniuk[r]
[s] Department of Statistics , Stanford University, Stanford, CA 94305
[r] Department of Electrical and Computer Engineering, Rice University, Houston, TX 77005
[p] Program in Applied and Computational Mathematics, Princeton University, Princeton, NJ 08544

Abstract

The emergence of low-cost sensing architectures for diverse modalities has made it possible to deploy sensor networks that acquire large amounts of very high-dimensional data. To cope with such a data deluge, manifold models are often developed that provide a powerful theoretical and algorithmic framework for capturing the intrinsic structure of data governed by a low-dimensional set of parameters.However, these models do not typically take into account dependencies among multiple sensors. We thus propose a new *joint manifold* framework for data ensembles that exploits such dependencies. We show that joint manifold structure can lead to improved performance for manifold learning. Additionally, we leverage recent results concerning random projections of manifolds to formulate a universal, network-scalable dimensionality reduction scheme that efficiently fuses the data from all sensors.

Introduction

The emergence of low-cost sensing devices has made it possible to deploy sensor networks that capture a single event from a large number of vantage points and using multiple modalities. This can lead to a veritable data deluge, fueling the need for efficient algorithms for processing and efficient protocols for transmitting the data generated by such networks. In order to address these challenges, there is a clear need for a theoretical framework for modeling the complex interdependencies among signals acquired by these networks. This framework should support the development of efficient algorithms that can exploit this structure and efficient protocols that can cope with the massive data volume.

Consider, for example, a sensor network consisting of J sensors simultaneously observing a common scene. We will assume that the signal acquired by each sensor is a length-N vector. Ideally, all sensors would send these raw signals to a central processing unit, which could then holistically analyze all the data produced by the network. This naïve approach would in general provide the best performance, since it exploits complete access to all of the data. However, the amount of raw data generated by a sensor network, on the order of JN, becomes untenably large even for fairly small networks operating at moderate sampling rates.

Alternatively, exploiting the fact that in many cases the end goal is to solve some kind of inference problem, each sensor could independently reach a decision or extract some relevant features, and then relay the result to the central processing unit which would then combine the results to provide the solution. Unfortunately, this approach also has disadvantages. The sensors must be "smart" in that they must possess some degree of sophistication so that they can execute nonlinear inference tasks. Such technology is expensive and can place severe demands on the available power resources. Second, the total power and bandwidth requirement scale *linearly* with the number of sensors J.

In order to cope with such high-dimensional data, a common strategy is to develop appropriate models for the acquired signals. A powerful model is the geometric notion of a low-dimensional *manifold*. Informally, manifold models arise in cases where *(i)* a K-dimensional parameter θ can be identified that carries the relevant information about a signal and *(ii)* the signal $f(\theta) \in \mathbb{R}^N$ changes as a continuous (typically nonlinear) function of these parameters. Typical examples include a one-dimensional (1-D) signal translated by an unknown time delay (parameterized by the translation variable) and an image of a 3-D object at an unknown location captured from an unknown viewing angle (parameterized by the three spatial coordinates of the object as well as its roll, pitch, and yaw). In these and many other cases, the geometry of the signal class forms a nonlinear K-dimensional manifold in $\mathbb{R}^N$,

$$\mathcal{M} = \{f(\theta) : \theta \in \Theta\}, \tag{1}$$

where Θ is the K-dimensional parameter space. In recent years, researchers in image processing have be-

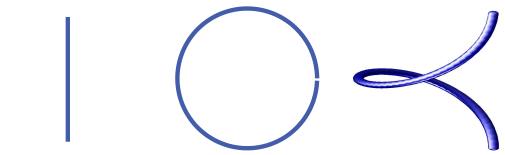
come increasingly interested in manifold models due to the observation that a collection of images obtained from different target locations/poses/illuminations and camera viewpoints form such a manifold [Lu1998, Donoho and Grimes2005, Wakin et al.2005]. As a result, manifold-based methods for image processing have attracted considerable attention, particularly in the machine learning community, and can be applied to diverse applications including data visualization, classification, estimation, detection, control, clustering, and learning [Wakin et al.2005, Belkin and Niyogi2003, Costa and Hero.2004].

In sensor networks, multiple observations of the same event are often acquired simultaneously, resulting in the acquisition of interdependent signals that share a common parameterization (such as the location and orientation of an object of interest). All of the acquired signals are functions of the same set of parameters, i.e., we can write each signal as $f_j(\theta)$ where $\theta \in \Theta$ is the same for all j. Our contention in this paper is that we can obtain a simple model that captures the correlation between the sensor observations by matching the parameter values for the different manifolds observed by the sensors. More precisely, we observe that by simply concatenating points that are indexed by the same parameter value θ from the different component manifolds, i.e., by forming $\boldsymbol{f}(\theta) = [f_1(\theta), f_2(\theta), \ldots, f_J(\theta)]$, we obtain a new manifold, which we dub the *joint manifold*, that encompasses all of the component manifolds and shares the same parameterization. This structure captures the interdependencies between the signals in a straightforward manner. We can then apply the same manifold-based processing techniques that have been proposed for individual manifolds to the entire ensemble of component manifolds.

As a key advantage of our proposed model, we illustrate how the joint manifold structure can be exploited via a simple and efficient data fusion algorithm based on *random projections*. For the case of J sensors jointly acquiring length-N signals sharing a common K-dimensional parameterization, we demonstrate that the total power and communication bandwidth required by our scheme is linear in the manifold dimension K and only *logarithmic* in the number of sensors J and the sensor resolution N. Recent developments in the field of compressive sensing has made this data acquisition model practical in many interesting applications [Donoho2006, Candès2006].

Joint Manifolds: Theory

In this section we develop a theoretical framework for ensembles of manifolds that are *jointly* parameterized by a small number of *common* degrees of freedom. Informally, we propose a data structure for jointly modeling such ensembles; this is obtained simply by concatenating points from different ensembles that are indexed by the same articulation parameter to obtain a single point in a higher-dimensional space.



(a) $\mathcal{M}_1 \subseteq \mathbb{R}$: line segment (b) $\mathcal{M}_2 \subseteq \mathbb{R}^2$: circle segment (c) $\mathcal{M}^* \subseteq \mathbb{R}^3$: helix segment

Figure 1: *A pair of isomorphic manifolds $\mathcal{M}_1$ and $\mathcal{M}_2$, and the resulting joint manifold $\mathcal{M}^*$.*

We begin by defining the joint manifold for the setting of general topological manifolds. In order to simplify our notation, we will let $\boldsymbol{\mathcal{M}} = \mathcal{M}_1 \times \mathcal{M}_2 \times \cdots \times \mathcal{M}_J$ denote the *product manifold*. Furthermore, we will use the notation $\boldsymbol{p} = (p_1, p_2, \ldots, p_J)$ to denote a J-tuple of points, or concatenation of J points, which lies in the Cartesian product of J sets (e.g., $\boldsymbol{\mathcal{M}}$).

Definition 1. *Let $\{\mathcal{M}_j\}_{j=1}^J$ be an ensemble of J topological manifolds of equal dimension K. Suppose that the manifolds are homeomorphic to each other, in which case there exists a homeomorphism ψ_j between $\mathcal{M}_1$ and $\mathcal{M}_j$ for each j. For a particular set of $\{\psi_j\}_{j=2}^J$, we define the* **joint manifold** *as*

$$\mathcal{M}^* = \{\boldsymbol{p} \in \boldsymbol{\mathcal{M}} : p_j = \psi_j(p_1), 2 \leq j \leq J\}.$$

Furthermore, we say that $\{\mathcal{M}_j\}_{j=1}^J$ are the corresponding **component manifolds**.

Note that $\mathcal{M}_1$ serves as a common *parameter space* for all the component manifolds. Since the component manifolds are homeomorphic to each other, this choice is ultimately arbitrary. In practice it may be more natural to think of each component manifold as being homeomorphic to some fixed K-dimensional parameter space Θ. However, in this case one could still define $\mathcal{M}^*$ as is done above by defining ψ_j as the composition of the homeomorphic mappings from $\mathcal{M}_1$ to Θ and from Θ to $\mathcal{M}_j$.

As an example, consider the one-dimensional manifolds in Fig. 1. Figures 1(a) and (b) show two isomorphic manifolds, where $\mathcal{M}_1 = (0, 2\pi)$ is an open interval, and $\mathcal{M}_2 = \{\psi_2(\theta) : \theta \in \mathcal{M}_1\}$ where $\psi_2(\theta) = (\cos(\theta), \sin(\theta))$, i.e., $\mathcal{M}_2 = S^1 \backslash (1, 0)$ is a circle with one point removed (so that it remains isomorphic to a line segment). In this case the joint manifold $\mathcal{M}^* = \{(\theta, \cos(\theta), \sin(\theta)) : \theta \in (0, 2\pi)\}$, illustrated in Fig. 1(c), is a helix. Notice that there exist other possible homeomorphic mappings from $\mathcal{M}_1$ to $\mathcal{M}_2$, and that the precise structure of the joint manifold as a submanifold of $\mathbb{R}^3$ is heavily dependent on the choice of this mapping.

Returning to the definition of $\mathcal{M}^*$, observe that although we have called $\mathcal{M}^*$ the joint manifold, we have

not shown that it actually forms a topological manifold. To prove that $\mathcal{M}^*$ is indeed a manifold, we will make use of the fact that the joint manifold is a subset of the product manifold $\boldsymbol{\mathcal{M}}$. One can show that $\boldsymbol{\mathcal{M}}$ forms a JK-dimensional manifold using the product topology [Boothby2003]. By comparison, we can show that $\mathcal{M}^*$ has dimension only K; see [Davenport et al.2010] for a proof. Since $\mathcal{M}^*$ is a submanifold of $\boldsymbol{\mathcal{M}}$, it also inherits some desirable properties from $\{\mathcal{M}_j\}_{j=1}^J$.

Proposition 1. *Suppose that $\{\mathcal{M}_j\}_{j=1}^J$ are isomorphic topological manifolds and $\mathcal{M}^*$ is defined as above. If $\{\mathcal{M}_j\}_{j=1}^J$ are Riemannian, then $\mathcal{M}^*$ is Riemannian.*

Up to this point we have considered general topological manifolds. In particular, we have *not* assumed that the component manifolds are embedded in any particular space. If each component manifold $\mathcal{M}_j$ is embedded in $\mathbb{R}^{N_j}$, the joint manifold is naturally embedded in $\mathbb{R}^{N^*}$ where $N^* = \sum_{j=1}^J N_j$. Hence, the joint manifold can be viewed as a model for sets of data with *varying ambient dimension* linked by a common parametrization. In the sequel, we assume that each manifold $\mathcal{M}_j$ is embedded in $\mathbb{R}^N$, which implies that $\mathcal{M}^* \subset \mathbb{R}^{JN}$. Observe that while the intrinsic dimension of the joint manifold remains constant at K, the ambient dimension increases by a factor of J.

We now examine how a number of geometric properties of the joint manifold compare to those of the component manifolds. We begin with the following simple observation that Euclidean distances[1] between points on the joint manifold are larger than distances on the component manifolds. The result follows directly from the definition of the Euclidean norm, so we omit the proof.

Proposition 2. *Let $\boldsymbol{p}, \boldsymbol{q} \in \mathcal{M}^*$ be given. Then*

$$\|\boldsymbol{p} - \boldsymbol{q}\| = \sqrt{\sum_{j=1}^J \|p_j - q_j\|^2}.$$

While Euclidean distances are important (especially when noise is introduced), the natural measure of distance between a pair of points on a Riemannian manifold is not Euclidean distance, but rather the *geodesic distance*. The geodesic distance between points $p, q \in \mathcal{M}$ is defined as

$$d_{\mathcal{M}}(p,q) = \inf_{\gamma}\{L(\gamma) : \gamma(0) = p, \gamma(1) = q\}, \quad (2)$$

where $\gamma : [0,1] \to \mathcal{M}$ is a C^1-smooth curve joining p and q, and $L(\gamma)$ is the length of γ as measured by

$$L(\gamma) = \int_0^1 \|\dot{\gamma}(t)\| dt. \quad (3)$$

[1]In the remainder of this paper, whenever we use the notation $\|\cdot\|$ we mean $\|\cdot\|_{\ell_2}$, i.e., the ℓ_2 (Euclidean) norm on $\mathbb{R}^N$. When we wish to differentiate this from other ℓ_p norms, we will be explicit.

We show geodesic distances on $\mathcal{M}^*$ compare to geodesic distances on the component manifolds via the following theorem (proven in [Davenport et al.2010]).

Theorem 1. *Suppose that $\{\mathcal{M}_j\}_{j=1}^J$ are Riemannian manifolds. Let $\boldsymbol{p}, \boldsymbol{q} \in \mathcal{M}^*$ be given. Then*

$$d_{\mathcal{M}^*}(\boldsymbol{p}, \boldsymbol{q}) \geq \frac{1}{\sqrt{J}} \sum_{j=1}^J d_{\mathcal{M}_j}(p_j, q_j). \quad (4)$$

If the mappings $\psi_2, \psi_3, \ldots, \psi_J$ are isometries, *i.e., $d_{\mathcal{M}_1}(p_1, q_1) = d_{\mathcal{M}_j}(\psi_j(p_1), \psi_j(q_1))$ for any j and for any pair of points $(\boldsymbol{p}, \boldsymbol{q})$, then*

$$d_{\mathcal{M}^*}(\boldsymbol{p}, \boldsymbol{q}) = \frac{1}{\sqrt{J}} \sum_{j=1}^J d_{\mathcal{M}_j}(p_j, q_j) = \sqrt{J} \cdot d_{\mathcal{M}_1}(p_1, q_1). \quad (5)$$

Next, we study local smoothness and global self avoidance properties of the joint manifold.

Definition 2. [Niyogi, Smale, and Weinberger2004] *Let $\mathcal{M}$ be a Riemannian submanifold of $\mathbb{R}^N$. The* **condition number** *is defined as $1/\tau$, where τ is the largest number satisfying the following: the open normal bundle about $\mathcal{M}$ of radius r is embedded in $\mathbb{R}^N$ for all $r < \tau$.*

The condition number controls both local smoothness properties and global properties of the manifold; as $1/\tau$ becomes smaller, the manifold becomes smoother and more self-avoiding. We wish to show that if the component manifolds are smooth and self avoiding, the joint manifold is as well. It is not easy to prove this in the most general case, where the only assumption is that there exists a homeomorphism (i.e., a continuous bijective map ψ) between every pair of manifolds. However, suppose the manifolds are *diffeomorphic*, i.e., there exists a continuous bijective map between tangent spaces at corresponding points on every pair of manifolds. In that case, we make the following assertion, proven in [Davenport et al.2010].

Theorem 2. *Suppose that $\{\mathcal{M}_j\}_{j=1}^J$ are Riemannian submanifolds of $\mathbb{R}^N$, and let $1/\tau_j$ denote the condition number of $\mathcal{M}_j$. Suppose also that the $\{\psi_j\}_{j=2}^J$ that define the corresponding joint manifold $\mathcal{M}^*$ are diffeomorphisms. If $1/\tau^*$ is the condition number of $\mathcal{M}^*$, then*

$$\frac{1}{\tau^*} \leq \max_{1 \leq j \leq J} \frac{1}{\tau_j}.$$

This result states that for general manifolds, the most we can say is that the condition number of the joint manifold is guaranteed to be less than that of the *worst* manifold. However, in practice this is not likely to happen. As an example, Fig. 2 illustrates the point at which the normal bundle intersects itself for the case of the joint manifold from Fig. 1(c). In this case we obtain

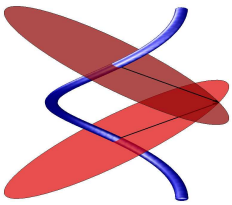


Figure 2: *Point at which the normal bundle for the helix manifold from Fig. 1(c) intersects itself. Note that the helix has been slightly rotated.*

$\tau^* = \sqrt{\pi^2/2 + 1} > 1$. Note that the condition numbers for the manifolds $\mathcal{M}_1$ and $\mathcal{M}_2$ generating $\mathcal{M}^*$ are given by $\tau_1 = \infty$ and $\tau_2 = 1$. Thus, while the condition number $1/\tau^*$ of the joint manifold is not as low as the best manifold, it is notably smaller than that of the worst manifold. In general, even this example may be somewhat pessimistic and it is possible that the joint manifold may be better conditioned than even the best manifold.

Joint Manifolds: Practice

The theory developed in the previous section suggests that the joint manifold preserves or improves the geometric properties of the component manifolds. In the next section we show that it can be extremely beneficial to use manifold learning algorithms specifically designed to exploit the joint manifold structure. However, we must first address some key practical concerns.

Acceptable deviations from theory

While manifolds are a natural way to model the structure of a set of signals governed by a small number of parameters, the results in the previous section make a number of assumptions concerning the structure of the component manifolds. In the most general case, we assume that the component manifolds are homeomorphic to each other. Such an assumption assures that the joint manifold is indeed a topological manifold.

Unfortunately, this excludes scenarios where a sensor network features non-overlapping fields of view. In such scenarios, there are cases in which only some sensors are sensitive to small changes in the parameter values. Strictly speaking, our theory may not apply in these cases, since the joint "manifold" as we have defined it is not necessarily even a topological manifold. We provide additional discussion of this issue in the sequel.

In our theoretical results concerning condition number, we also assume that the component manifolds are smooth. However, the image manifolds induced by the motion of an object where there are sharp edges or occlusions are nowhere differentiable. In a camera network, this problem can be addressed by applying a smoothing kernel to each captured image, inducing a smooth manifold [Wakin et al.2005].

Efficient data fusion via joint manifolds using linear projections

Observe that when the number J and ambient dimension N of the manifolds become large, the ambient dimension of the joint manifold—JN—may be so large that it becomes impossible to perform any meaningful computations. Furthermore, it might appear that in order to exploit the joint manifold structure, we must collect all the data at a central location, which we earlier claimed was potentially impossible.

To address this challenge, we suppose that we are given a network of J sensors, let $x_j \in \mathbb{R}^N$ denote the signal acquired by sensor j, which is assumed to belong in a manifold $\mathcal{M}_j$, and let $\boldsymbol{x}$ denote the corresponding point in the joint manifold $\mathcal{M}^*$. Observe that if we had access to the vector $\boldsymbol{x}$, then we could exploit the joint manifold structure to map it to a parameter vector $\widehat{\theta}$ of length only K rather than JK. Unfortunately, this mapping will generally be nonlinear, and each element of $\widehat{\theta}$ could potentially depend on the entire vector $\boldsymbol{x}$, preventing us from operating individually on each x_j. Thus, rather than directly extract the features, we will instead restrict our focus to *linear* dimensionality reduction methods that, while acting on the concatenated data $\boldsymbol{x}$, can be implemented in a distributed fashion.

Specifically, we will aim to compute a dimensionally reduced representation of $\boldsymbol{x}$ denoted $\boldsymbol{y} = \boldsymbol{\Phi x}$, where $\boldsymbol{\Phi}$ is a standard linear projection operator. Since the operator is linear, we can take *local* projections of the signals acquired by each sensor, and still calculate the *global* projections of $\boldsymbol{x}$ in a distributed fashion. Let each sensor calculate $y_j = \Phi_j x_j$, with the matrices $\Phi_j \in \mathbb{R}^{M \times N}, 1 \leq j \leq J$. Then, by defining the $M \times JN$ matrix $\boldsymbol{\Phi} = [\Phi_1 \ \Phi_2 \ \cdots \ \Phi_J]$, the global projections $\boldsymbol{y} = \boldsymbol{\Phi x}$ can be obtained by

$$\begin{aligned}\boldsymbol{y} = \boldsymbol{\Phi x} &= [\Phi_1 \quad \Phi_2 \quad \cdots \quad \Phi_J][x_1^T \quad x_2^T \quad \cdots \quad x_J^T]^T \\ &= \Phi_1 x_1 + \Phi_2 x_2 + \cdots + \Phi_J x_J.\end{aligned}$$

Thus, the final measurement vector can be obtained by simply *adding independent projections* of the signals acquired by the individual sensors. This gives rise to a *compressive data fusion* protocol.

The main challenge in designing such a scheme is the choice of a suitable matrix $\boldsymbol{\Phi}$ for a specific joint manifold $\mathcal{M}^*$ that preserves its Euclidean and the geodesic structures while ensuring that M is comparable to the dimension K of the joint manifold (and hence much less than the ambient dimension JN). Fortunately, we can exploit recent results concerning *random projections* to solve this problem without any prior knowledge of the structure of the network or the objects to be captured. Specifically, it has been shown that the essential structure of a K-dimensional manifold with condition number $1/\tau$ residing in $\mathbb{R}^N$ is approximately preserved under an orthogonal projection into a random

subspace of dimension $O(K\log(N/\tau)) \ll N$ [Baraniuk and Wakin2009]. Thus, we obtain a faithful embedding of the joint manifold via a representation of dimension only $O(K\log JN)$.

Joint manifold fusion via random projections, like compressive sensing [Donoho2006,Candès2006,Duarte et al.2008], is *universal* in that the measurement process is not dependent on the specific structure of the manifold. Thus, our sensing techniques need not be replaced for these extensions; only our underlying models (hypotheses) are updated.

Joint Manifold Learning

In this section we demonstrate that in a variety of settings, the joint manifold is significantly easier to learn than the individual component manifolds. This improvement is due to both the kind of increased robustness to noise described earlier and to the fact that, as was shown in Theorem 2, the joint manifold can be significantly better-conditioned than the component manifolds, meaning that it is easier to learn the structure of the joint manifold from a finite sampling of points.

Theory

Several algorithms for manifold learning have been proposed, each giving rise to a nonlinear map with its own special properties and advantages (e.g., Isomap [Tenenbaum, Silva, and Landford2000], Locally Linear Embedding (LLE) [Roweis and Saul2000], Hessian Eigenmaps [Donoho and Grimes2003], etc.) Of these approaches, we devote special attention here to the Isomap algorithm, which assumes that the point cloud consists of samples from a data manifold that is (at least approximately) isometric to a convex subset of Euclidean space.

Isomap works in three stages:

1. Construct a graph G that contains a vertex for each data point; an edge connects two vertices if the Euclidean distance between the corresponding data points is below a specified threshold.
2. Weight each edge in the graph G by computing the Euclidean distance between the corresponding data points. We then estimate the geodesic distance between each pair of vertices as the length of the shortest path between the corresponding vertices in the graph G.
3. Embed the points in $\mathbb{R}^K$ using multidimensional scaling (MDS), which attempts to embed the points so that their Euclidean distance approximates the estimated geodesic distances.

We examine the performance of Isomap for learning the joint manifold as compared to learning the J isometric component manifolds separately. We assume that we have noiseless samples from $\{\mathcal{M}_j\}_{j=1}^J$. In order to judge the quality of the embedding learned by Isomap, we will observe that for any pair of samples p, q from a manifold $\mathcal{M}$ whose vertices are linked within the graph G, we have that

$$\rho \leq \frac{\|p-q\|}{d_{\mathcal{M}}(p,q)} \leq 1 \tag{6}$$

for some $\rho \in [0,1]$ that will depend on the samples of $\mathcal{M}$ and the graph G. Isomap will perform well if the largest value of ρ that satisfies (6) for any pair of samples that are connected by an edge in the graph G is close to 1. Using this fact, we can compare the performance of manifold learning using Isomap on samples from the joint manifold $\mathcal{M}^*$ to using Isomap on samples from a particular component manifold $\mathcal{M}_k$. The proof of this theorem can be found in [Davenport et al.2009].

Theorem 3. *Let $\mathcal{M}^*$ be a joint manifold from J isometric component manifolds. Let $\boldsymbol{p}, \boldsymbol{q} \in \mathcal{M}^*$ and suppose that we are given a graph G that contains one vertex for each sample obtained from $\mathcal{M}^*$. For each $k = 1, \ldots, J$, define ρ_j as the largest value such that*

$$\rho_j \leq \frac{\|p_j - q_j\|}{d_{\mathcal{M}_j}(p_j, q_j)} \leq 1 \tag{7}$$

for all pairs of points connected by an edge in G. Then we have that

$$\frac{1}{\sqrt{J}}\sqrt{\sum_{j=1}^{J}\rho_j^2} \leq \frac{\|\boldsymbol{p}-\boldsymbol{q}\|}{d_{\mathcal{M}^*}(\boldsymbol{p},\boldsymbol{q})} \leq 1. \tag{8}$$

From Theorem 3 we see that, in many cases, the joint manifold estimates of the geodesic distances will be more accurate than the estimates obtained using one of the component manifolds. If for a particular component manifold $\mathcal{M}_k$ we observe that $\rho_k \leq \sqrt{\sum_{j=1}^J \rho_j^2/J}$, then we know that the joint manifold leads to better estimates. Essentially, we may expect that the joint manifold will lead to estimates that are better than the average case across the component manifolds.

We now consider the case where we have a dense sampling of the manifolds so that the $\rho_j \approx 1$, and examine the case where we obtain noisy samples. We will assume that the noise is i.i.d. and demonstrate that any distance calculation performed on $\mathcal{M}^*$ serves as a better estimator of the pairwise (and consequently, geodesic) distances between any two points $\boldsymbol{p}$ and $\boldsymbol{q}$ than that performed on any component manifold using the points p_j and q_j. Again, the proof of this theorem can be found in [Davenport et al.2009].

Theorem 4. *Let $\mathcal{M}^*$ be a joint manifold from J isometric component manifolds. Let $\boldsymbol{p}, \boldsymbol{q} \in \mathcal{M}^*$ and assume that $\|p_j - q_j\| = d$ for all j. Assume that we acquire noisy observations $\boldsymbol{s} = \boldsymbol{p} + \boldsymbol{n}$ and $\boldsymbol{r} = \boldsymbol{q} + \boldsymbol{n}'$, where $\boldsymbol{n}$ and $\boldsymbol{n}'$ are independent noise vectors with*

$\mathbb{E}[\|n_j\|^2] = \mathbb{E}[\|n'_j\|^2] = \sigma^2$, $\|n_j\|^2 \leq \epsilon$, *and* $\|n'_j\|^2 \leq \epsilon$ *for* $j = 1, \ldots, J$. *Then,*

$$P\left(1 - \delta \leq \frac{\|\boldsymbol{s} - \boldsymbol{r}\|^2}{\|\boldsymbol{p} - \boldsymbol{q}\|^2 + 2J\sigma^2} \leq 1 + \delta\right) \geq 1 - 2c^{-J^2},$$

where $c = \exp\left(2\delta^2 \left(\frac{d^2 + 2\sigma^2}{d\sqrt{\epsilon} + \epsilon}\right)^2\right)$.

We observe that the estimate of the true distance suffers from a small constant bias; this can be handled using a simple debiasing step.[2] Theorem 4 indicates that the probability of large deviations in the estimated distance decreases *exponentially* in the number of component manifolds J; thus we should observe significant "denoising" even in the case where J is relatively small.

Practice

Our theoretical results assume that the acquired data arises from J isometric component manifolds. As noted earlier, barring controlled or synthetic scenarios, this is very rarely the case. In practice, the isometric assumption breaks down due to two reasons: *(i)* the sensors may be at different distances from the scene, non-identical sensors may possess different dynamic ranges, or the sensors may be of different modalities (such as visual versus infrared cameras or even visual plus audio data), and thus the component manifolds may be scaled differently; *(ii)* in a camera networks there may be occlusions or partially-overlapping fields of view, and certain regions of component manifolds may be ill-conditioned.

In order to handle such non-idealities, we make two modifications to the Isomap algorithm while performing joint manifold learning. Recall that that in order to find the nearest-neighbor graph G, Isomap must first calculate the matrix of squared pairwise Euclidean distances. We denote this matrix $\mathbf{D}$ for the joint manifold $\mathcal{M}^*$ and D_j for the component manifold $\mathcal{M}_j$. Note that $\mathbf{D} = \sum_{j=1}^{J} D_j$. Thus, if a particular component manifold is scaled differently than the others, by which we mean that $d_{\mathcal{M}_j}(f_j(\theta_1), f_j(\theta_2)) = C_j\|\theta_1 - \theta_2\|_2$ with $C_j \neq 1$, then all the entries of the corresponding D_j will be reweighted by C_j^2, so that D_j will have a disproportionate impact on $\mathbf{D}$. This corresponds to the first non-ideality described above, and can be alleviated by *normalizing* each D_j by its Frobenius norm, which can be interpreted as scaling each manifold so that an Eulerian path through the complete graph has unit length.

The second non-ideality can be partially addressed by attempting to adaptively detect and correct for occlusion events. Consider the case of large-scale occlusions, in which we make the simplifying assumption that for each camera the object of interest is either entirely within the camera's view or entirely occluded. In this case, the non-occluded component manifolds are still locally isometric to each other, i.e., there exists a neighborhood U such that $d_{\mathcal{M}_j}(f_j(\theta_1), f_j(\theta_2)) = \|\theta_1 - \theta_2\|_2$ for all $\theta_1, \theta_2 \in U$ and for all j corresponding to the non-occluded component manifolds. Thus, if we knew which cameras were occluded for a pair of points, say $\boldsymbol{x}^m$ and $\boldsymbol{x}^n$, then we could simply ignore those cameras in computing $\mathbf{D}_{m,n}$ and rescale $\mathbf{D}_{m,n}$ so that it is comparable with the case when no cameras exhibit occlusions. More specifically, we let $\widetilde{J}$ denote the index set for non-occluded component manifolds and set $\mathbf{D}_{m,n} = (|J|/|\widetilde{J}|)\sum_{j \in \widetilde{J}} \|x_j^m - x_j^n\|_2^2$. To do this automatically, we compare $\|x_j^m - x_j^n\|_2^2$ to a specified threshold, i.e., we set $\widetilde{J} = \{j : \|x_j^m - x_j^n\|_2^2 \geq \epsilon\}$ for some parameter ϵ, since for the component manifolds in which the object of interest is occluded this distance will be relatively small. The parameter ϵ can be reasonably inferred from the data. $\mathbf{D}$ is used by subsequent steps in Isomap to learn an improved low-dimensional embedding of the high-dimensional acquired data. Note that while this approach does not rigorously handle boundary cases where objects are only partially occluded, our experimental results below indicate that the algorithms are robust to such cases. Such an approach could be readily applied to other sensor networks which experience occlusion-like phenomena.

Experiments

We provide some results using data from a camera network that demonstrate the significant gains obtained by exploiting the joint manifold structure, both with and without the use of random projections. The images are obtained from a network of four Unibrain Fire-i™ OEM Firewire board cameras. Each camera has resolution $N = 320 \times 240 = 76800$ pixels. The manifold learning results have been generated using Isomap. All of our experiments are performed on 2-D image manifolds isomorphic to a closed rectangular subset of $\mathbb{R}^2$.

Learning with occlusions In this experiment we study the impact of occlusions in a camera network. The data comprises $J = 4$ different views of the independent motions of 2 toy koalas along individual 1-D paths, yielding a 2-D combined parameter space. This data suffers from real-world artifacts such as fluctuations in illumination conditions and variations in the pose of the koalas; further, the koalas occlude one another in certain views or are absent from certain views depending on the particular vantage point. Sample images and 2-D embedding results are displayed in Figure 3. We observe that the best embedding is obtained by using the modified version of Isomap for learning the

[2] Manifold learning algorithms such as Isomap deal with biased estimates of distances by "centering" the matrix of squared distances, i.e., removing the mean of each row/column from every element.

joint manifold. We compute $M = 2400$ random projections of each image and sum them to obtain a randomly projected version of the joint data and repeat the above experiment. The dimensionality of the projected data is only 3% of the original data; yet, we see very little degradation in performance, thus displaying the effectiveness of random projection-based fusion.

Unsupervised target tracking As a practical application of manifold learning, we consider a situation where we are given a set of training data consisting of images of a target moving through a region along with a set of test images of the target moving along a particular trajectory. We do not explicitly incorporate any known information regarding the locations of the cameras or the parameter space describing the target's motion. The training images comprise $J = 4$ views of a coffee mug placed at different positions on an irregular rectangular grid. Example images from each camera are shown in Figure 4. For the test data, we translate the coffee mug so that its 2-D path traces out the shape of the letter "R". We aim to recover this shape using both the test and training data. To solve this problem, we attempt to learn a 2-D embedding of the joint manifold using the modified version of Isomap detailed earlier. The learned embedding for each camera is shown in Figure 4. As is visually evident, learning the data using any one camera yields very poor results; however learning the joint manifold helps discern the 2-D structure to a much better degree. In particular, the "R" trajectory in the test data is correctly recovered only by learning the joint manifold. Finally, we repeat the above procedure using $M = 4800$ random projections of each image, and fuse the data by summing the measurement vectors. While the recovered trajectory of the anomalous (test) data suffers some degradation in visual quality, we observe comparable 2-D embedding results for the individual and joint manifolds as with the original data set. Since the dimensionality of the projected data is merely 6% that of the original data set, this would translate to significant savings in communication costs in a real-world camera network.

Discussion

Joint manifolds naturally capture the structure present in the data produced by sensor networks. We have studied topological and geometric properties of joint manifolds, and have provided some basic examples that illustrate how they can improve the performance of common signal processing algorithms. We have also introduced a simple framework for data fusion for sensor networks that employs independent random projections of each signal, which are then accumulated to obtain an accurate low-dimensional representation of the joint manifold. Our fusion scheme can be directly applied to the data acquired by such devices.

Acknowledgements This work was supported by grants NSF CCF-0431150 and CCF-0728867, DARPA/ONR N66001-08-1-2065, ONR N00014-07-1-0936 and N00014-08-1-1112, AFOSR FA9550-07-1-0301 and FA9550-09-1-0432, ARO MURIs W911NF-07-1-0185 and W911NF-09-1-0383, and the Texas Instruments Leadership University Program.

References

Baraniuk, R., and Wakin, M. 2009. Random projections of smooth manifolds. *Found. Comput. Math.* 9(1):51–77.

Belkin, M., and Niyogi, P. 2003. Using manifold structure for partially labelled classification. In *Advances in NIPS*, volume 15. MIT Press.

Boothby, W. 2003. *An Introduction to Differentiable Manifolds and Riemannian Geometry*. London, England: Academic Press.

Candès, E. J. 2006. Compressive sampling. In *Proc. Int. Cong. Math.*, volume 3, 1433–1452.

Costa, J., and Hero., A. 2004. Geodesic entropic graphs for dimension and entropy estimation in manifold learning. *IEEE Trans. Signal Proc.* 52(8):2210–2221.

Davenport, M.; Hegde, C.; Duarte, M.; and Baraniuk, R. 2009. A theoretical analysis of joint manifolds. Technical Report TREE0901, Rice University ECE Department.

Davenport, M.; Hegde, C.; Duarte, M.; and Baraniuk, R. 2010. Joint manifolds for data fusion. *to appear in IEEE Trans. Image Proc.*

Donoho, D., and Grimes, C. 2003. Hessian eigenmaps: locally linear embedding techniques for high dimensional data. *Proc. National Academy of Sciences* 100(10):5591–5596.

Donoho, D., and Grimes. 2005. Image manifolds which are isometric to Euclidean space. *J. Math. Imaging and Computer Vision* 23(1).

Donoho, D. 2006. Compressed sensing. *IEEE Trans. Info. Theory* 52(4):1289–1306.

Duarte, M.; Davenport, M.; Takhar, D.; Laska, J.; Sun, T.; Kelly, K.; and Baraniuk, R. 2008. Single pixel imaging via compressive sampling. *IEEE Signal Proc. Mag.* 25(2):83–91.

Lu, H. 1998. *Geometric Theory of Images*. Ph.D. Dissertation, University of California, San Diego.

Niyogi, P.; Smale, S.; and Weinberger, S. 2004. Finding the homology of submanifolds with confidence from random samples. Technical Report TR-2004-08, University of Chicago.

Roweis, S., and Saul, L. 2000. Nonlinear dimensionality reduction by locally linear embedding. *Science* 290:2323–2326.

Tenenbaum, J.; Silva, V.; and Landford, J. 2000. A global geometric framework for nonlinear dimensionality reduction. *Science* 290:2319–2323.

Wakin, M.; Donoho, D.; Choi, H.; and Baraniuk, R. 2005. The multiscale structure of non-differentiable image manifolds. In *Proc. Wavelets XI at SPIE Optics and Photonics*.

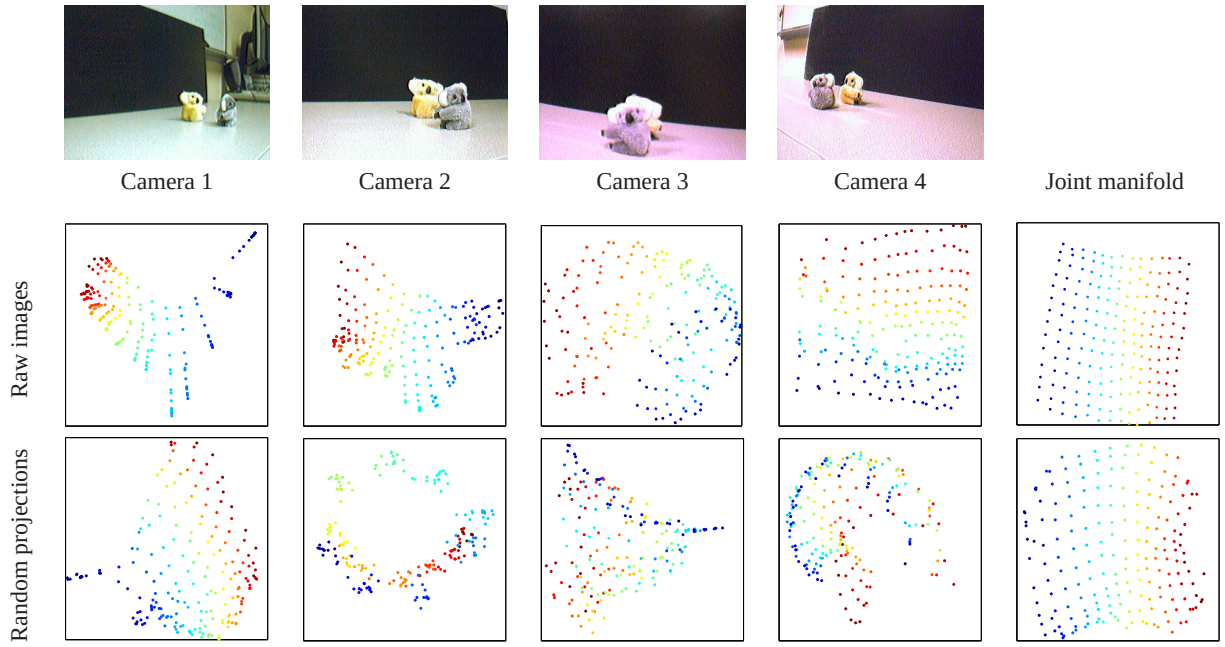


Figure 3: *(top) Sample images of 2 koalas moving along individual 1-D paths, yielding a 2-D manifold; (middle) 2-D embeddings of the dataset learned via Isomap from $N = 76800$ pixel images; (bottom) 2-D embeddings of the dataset learned from $M = 2400$ random projections. Learning the joint manifold yields a much improved 2-D embedding.*

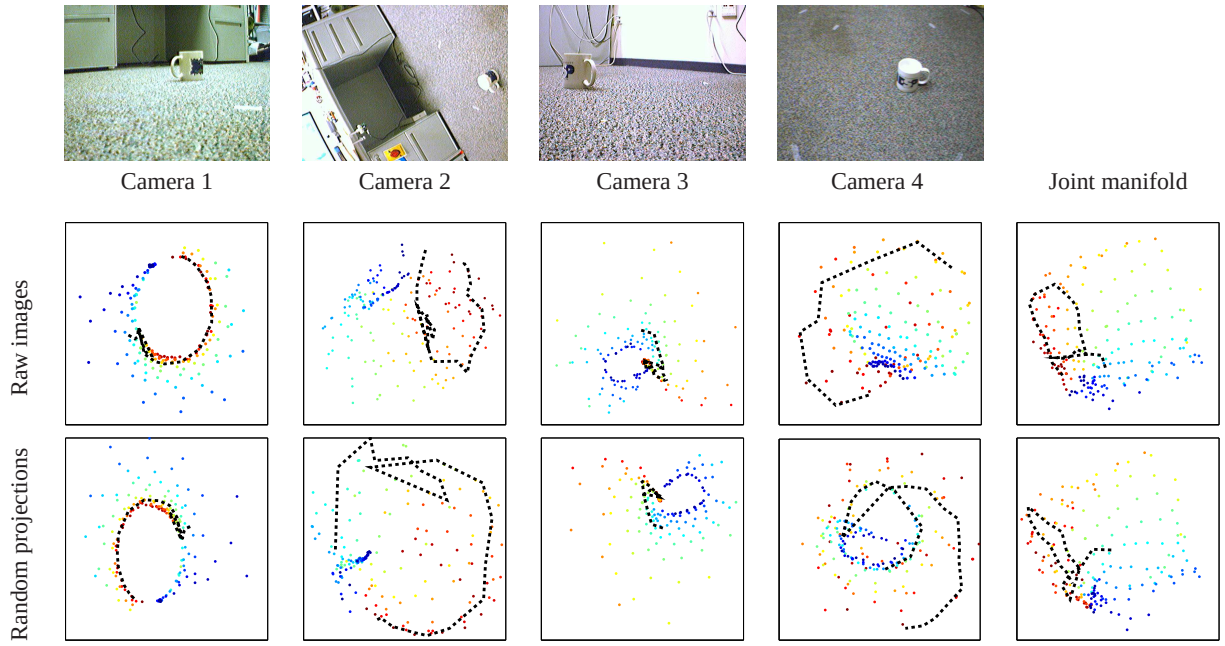


Figure 4: *(top) Sample images of the 2-D movement of a coffee mug; (middle) 2-D embeddings of the dataset learned via Isomap from $N = 76800$ pixel images; (bottom) 2-D embeddings of the dataset learned via Isomap from $M = 4800$ random projections. The black dotted line corresponds to an "R"-shaped trajectory in physical space. Learning the joint manifold yields a much improved 2-D embedding of the training points, as well as the "R"-shaped trajectory.*