IEEE Robotics and Automation Letters (RAL) paper presented at the
2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

TSBP: Tangent Space Belief Propagation for Manifold Learning

Thomas Cohn[1], Odest Chadwicke Jenkins[1], Karthik Desingh[1], Zhen Zeng[1]

***Abstract*— We present Tangent Space Belief Propagation (TSBP) as a method for graph denoising to improve the robustness of manifold learning algorithms. Dimension reduction by manifold learning relies heavily on the accurate selection of nearest neighbors, which has proven an open problem for sparse and noisy datasets. TSBP uses global nonparametric belief propagation to accurately estimate the tangent spaces of the underlying manifold at each data point. Edges of the neighborhood graph that deviate from the tangent spaces are then removed. The resulting denoised graph can then be embedded into a lower-dimensional space using methods from existing manifold learning algorithms. Artificially generated manifold data, simulated sensor data from a mobile robot, and high dimensional tactile sensory data are used to demonstrate the efficacy of our TSBP method.**

I. Introduction

We now have access to vast repositories of data with increasingly detailed images, point clouds, motion capture, tactile information, and other modes of robot sensing. Advances in machine learning have provided valuable tools and algorithms for the analysis of this wealth of robot data, but many existing machine learning algorithms can struggle when provided with small amounts of sparse, higher-dimensional data [1]. Problems can occur due to computational intractability as dimensionality and noise grow [2], [3], [4], [5], [6] as well as data inefficiency with opaque or inexplicable modes of failure [7], [8]. It remains an open problem how to generalize over sparse data as it grows in scale.

Manifold learning, a form of dimensionality reduction, is well-suited to address the scaling issues of robot data. Dimensionality reduction, in general, is the problem of finding lower-dimensional representations of data that preserve the apparent similarities and dissimilarities. Oftentimes, the data can be modeled as lying along a smooth manifold embedded in the higher-dimensional input space. Manifold learning is the problem of discovering this structure, and using it to perform dimensionality reduction. Manifold learning algorithms have been used in robotics for uncovering subspace structures across a variety of data modalities [9], [10], [11], [12], [13].

Many existing manifold learning methods use nearest-neighbors search to interpret the apparent lower dimensional structure [14], [15], [16]. The resulting neighborhood graph alone can provide enough information to embed the data into a lower-dimensional space; however, if distinct regions of the manifold pass close to each other and the data is sufficiently sparse or noisy, a nearest-neighbors search can find edges that incorrectly connect the distinct regions [17]. These "short-circuit" edges imply a close proximity relation between points that are actually distant in the underlying manifold. We say that such edges "cross" the manifold. The result causes existing algorithms to output an inaccurate embedding, where dissimilar data points are placed close together.

The goal of TSBP is to remove these erroneous edges. We use the fact that smooth manifolds are locally well-approximated by their tangents, and that even in the presence of noise, valid edges will be nearly contained within the nearby tangent spaces. In contrast, edges which cross the manifold disagree with the tangent spaces of their endpoints. By pruning away edges misaligned with manifold tangents, existing algorithms can produce a more accurate embedding.

The challenge lies in attempting to estimate the tangent space at a given point. Principal Component Analysis (PCA) [18] is often a reasonable choice for tangent space estimation, but it relies on local neighborhoods that may include nearby points from other sections of the manifold. As a result, inaccurate tangent space estimates will occur at the data points that are the most critical for embedding.

In this paper, we present Tangent Space Belief Propagation (TSBP) as a method for graph denoising to improve the robustness of manifold learning algorithms. TSBP aims to accurately estimate the tangent spaces of each point in the manifold via global belief propagation. To perform this belief propagation, TSBP employs an efficient "pull" approach to nonparametric message passing [19] suited to linear-time computation of message products. Convergence of this message passing results in accurate tangent space estimates, which are then used to remove edges which cross the manifold. The resulting denoised graph can then be embedded into a lower-dimensional space using methods from existing manifold learning algorithms, such as ISOMAP [14]. We demonstrate the improvement in accuracy from TSBP with synthetic "toy" data sets, a simulated robot mapping task, and an object classification experiment utilizing high-dimensional tactile data.

*This work was supported in part by NSF award IIS-1638060.

[1]Thomas Cohn, Odest Chadwicke Jenkins, Karthik Desingh, and Zhen Zeng are with the Department of Electrical Engineering and Computer Science and Engineering, Robotics Institute, University of Michigan, Ann Arbor, MI 48109, USA [cohnt|ocj|kdesingh|zengzhen]@umich.edu

II. Related Work

A. Dimensionality Reduction Algorithms

For problems where the data are embedded in a non-linear manifold, classical dimension reduction methods, such as PCA [18] and Multidimensional Scaling (MDS) [20], struggle to produce accurate low-dimensional embeddings. Many manifold learning algorithms rely on the construction of accurate local neighborhoods to approximate the manifold structure. ISOMAP [14] uses a nearest-neighbors search to approximate geodeisic distance along the manifold, and finds an embedding that best preserves the pairwise distances. Locally Linear Embedding (LLE) [15] constrains points to the estimated manifold tangent space that best approximates their local neighborhood, and learns a neighborhood-preserving mapping. Local Tangent Space Alignment (LTSA) [21] extends LLE by aligning the estimated tangent spaces, flattening out the manifold to a lower dimensional space. Laplacian Eigenmaps [16] uses a nearest-neighbors search, and weights connections between points using principles from manifold calculus. These algorithms are effective, and often yield similar results, as long as the local neighborhoods are accurate.

Unfortunately, when data are sparse or noisy, constructing accurate neighborhoods is challenging. The nearest-neighbors search may connect points from different regions of the manifold, creating short-circuit edges that cause the above algorithms to produce poor embeddings.

Other dimensionality reduction techniques do not assume the data lies along a manifold. t-SNE [22] uses pairwise distances to model the similarity of data points as joint probabilities, and then minimizes the Kullback-Leibler divergence to map the points to a lower-dimensional space. These distances can also be replaced with the geodeisic distances, computed along a neighborhood graph. Autoencoders [23] are a class of neural network that attempt to learn a lower-dimensional data representation while minimizing reconstruction error. When data does indeed lie along a manifold, they cannot take advantage of this additional structure, and may perform less effectively than their manifold learning counterparts. And when applied to sparse, noisy datasets, autoencoders may suffer from overfitting.

B. Noise Reduction Techniques

Because existing manifold learning algorithms rely upon an accurate neighborhood structure, recent research has been devoted to constructing more accurate neighborhood graphs. Local Smoothing [24] is an iterative process that seeks to concurrently handle outliers and reduce noise. Proximity Graphs [25] constructs a better graph for manifold learning by using an ensemble of minimum spanning trees. EIV Smoothing [26] assumes that all points in a local neighborhood are potentially noisy, and attempts to reconcile them according to the local geometry. Robust Hessian LLE (RHLLE) [27] is another variant of LLE that uses outlier removal and local smoothing. These algorithms are often effective, but they struggle with sparse data: regions with low density may be pulled apart or discarded altogether, disconnecting the neighborhood graph. The algorithms also may not be able to remove all of the short-circuit edges, just one of which can adversely affect the embedding.

BP-ISOMAP [17] is an extension of ISOMAP that includes a graph denoising step. After performing a nearest-neighbors search, the algorithm estimates edge weights via belief propagation, and edges with distances greater than a set threshold are removed from the graph. This algorithm is very effective at removing isolated short-circuit edges. However, BP-ISOMAP only utilizes edge lengths when performing belief propagation, so it struggles with sparse data. And because belief propagation is performed separately on each local neighborhood, it cannot handle when false edges are near each other.

III. TSBP: Tangent Space Belief Propagation

We begin by formulating the problems of manifold learning and tangent space estimation, before presenting our algorithm for neighborhood graph denoising.

A. Manifold Learning

The manifold learning problem is defined as follows. Let $\mathcal{M} \subseteq \mathbb{R}^n$ be a smooth, compact, connected k-dimensional manifold, with $1 \leq k \leq n$. We are given data $X = \{x_1, \ldots, x_N\} \subseteq \mathcal{M}$. The goal is to find $H = \{\eta_1, \ldots, \eta_N\} \subseteq \mathbb{R}^k$, with $f : \mathbb{R}^k \to \mathbb{R}^n$ defined by $f(\eta_i) = x_i$, $\forall \eta_i$, so that the η_i's preserve the perceived similarities and dissimilarities of the x_i's.

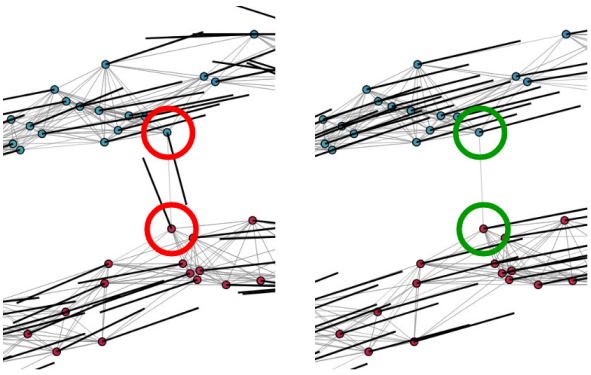
We extend the problem statement to account for noise by assuming we are instead given $\hat{x}_1, \ldots, \hat{x}_N \in \mathbb{R}^n$ with each $\hat{x}_i \approx x_i$. The distribution of the noise is unknown, but bounded by some $\varepsilon > 0$, i.e., $\forall i$, $||\hat{x}_i - x_i|| < \varepsilon$.

B. Manifold Geometry and Tangent Spaces

At each point $\hat{x}_i$, we have a corresponding affine tangent space $\mathcal{T}_{\hat{x}_i} \in (\mathbb{R}^n)^k$. Assume $\hat{x}_1$ is sufficiently close to $\hat{x}_2$; then the angle between the vector $\vec{v} = \hat{x}_2 - \hat{x}_1$ and $\mathcal{T}_{\hat{x}_i}$ should be small. If $\vec{v}$ does not agree with the tangent space, then $\hat{x}_1$ and $\hat{x}_2$ are falsely connected in the neighborhood graph. Thus, given the tangent spaces, we can determine which edges cross the manifold.

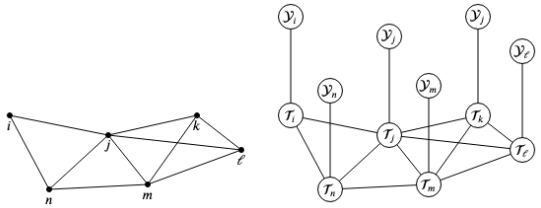
We cannot directly compute $\mathcal{T}_{\hat{x}_i}$ because we do not know the underlying manifold. A common approximation is performing PCA on local neighborhoods. Let $w_1, \ldots, w_k \in \mathbb{R}^n$ be the first k principal components of a neighborhood of x_i; then $\text{span}(w_1, \ldots, w_k) \approx \mathcal{T}_{\hat{x}_i}$. But if the neighborhood includes points from a different region of the manifold, PCA yields an inaccurate estimate, as shown in Figure 1.

The tangent bundle of a smooth manifold is itself a smooth manifold [28], so points which are sufficiently



(a) PCA **(b)** Ground Truth

Fig. 1: A comparison of the estimated tangent space from PCA with the ground truth. The red and blue points are from different regions of the manifold, but are close enough that there's an edge connecting them. Thin, grey edges are the neighborhood graph, and the thicker black lines are the tangent space estimates (the first component of the PCA). This noisy neighborhood graph leads to a poor tangent space estimate. Additionally observe that adjacent data points have very similar tangents in the ground truth.



(a) Neighborhood graph **(b)** Corresponding MRF

Fig. 2: An example of a MRF constructed from a neighborhood graph. The latent variable $\mathcal{T}_t$ represents the tangent space of the point t, and the corresponding observation is represented by $\mathcal{Y}_t$.

near to each other have similar tangent spaces. This means $\hat{x}_i$ and $\hat{x}_j$ should be neighbors only if $\mathcal{T}_{\hat{x}_i} \approx \mathcal{T}_{\hat{x}_j}$. If this expectation doesn't hold, we infer that $\hat{x}_i$ and $\hat{x}_j$ must be from different regions of the manifold.

C. Tangent Space Estimation

We form a belief of the tangent space at each point based on the principal components of the neighborhood and the beliefs of other points in the neighborhood. We formulate the problem of tangent space estimation as inference on a Markov Random Field (MRF). Let $G = (\mathcal{T} \sqcup \mathcal{Y}, E)$ be our graph. For a given point $\hat{x}_i$, let $\mathcal{T}_i \in \mathcal{T}$ be the latent variable representing its tangent space, and let $\mathcal{Y}_i \in \mathcal{Y}$ be its observation, as obtained from the k principal components of the neighborhood. $\forall i, j$, $(\mathcal{T}_i, \mathcal{Y}_i) \in E$, and $(\mathcal{T}_i, \mathcal{T}_j) \in E$ only if $\hat{x}_i$ and $\hat{x}_j$ are adjacent w.r.t. the neighborhood graph (Figure 2). Thus, the joint probability of G is

$$\mathbb{P}(\mathcal{T}, \mathcal{Y}) = \frac{1}{Z} \prod_{(i,j)\in E} \psi(\mathcal{T}_i, \mathcal{T}_j) \prod_{i \in V} \phi(\mathcal{T}_i, \mathcal{Y}_i) \quad (1)$$

where ψ is a pairwise potential function representing the compatibility between two neighboring tangent spaces $\mathcal{T}_i$ and $\mathcal{T}_j$, ϕ is a unary potential function representing the compatibility between an estimated tangent space $\mathcal{T}_i$ and its observation $\mathcal{Y}_i$, and Z is a normalizing factor. Although the PCA observation may be inaccurate, and the neighborhood may contain points from different regions of the manifold, we are still able to infer an accurate estimate of the tangent space.

Because both the unary and pairwise potential functions involve comparing tangent spaces, we need some metric to measure the dissimilarity of a pair of arbitrary vector spaces. For each basis vector of one subspace, we project it onto the other subspace, and measure the magnitude of the difference between the original and projected vector. The sum of these magnitudes describes how much the spaces differ. Formally, let $U = (u_1, \ldots, u_k), V = (v_1, \ldots, v_k) \in (\mathbb{R}^n)^k$. We define a dissimilarity metric

$$\Gamma(U, V) = \sum_{i=1}^{k} ||u_i - \text{proj}_V u_i||^2 \quad (2)$$

This leads to our unary and pairwise potentials. For a given $\mathcal{T}_i$ and $\mathcal{Y}_i$, the unary potential is

$$\phi(\mathcal{T}_i, \mathcal{Y}_i) = (1 + \Gamma(\mathcal{T}_i, \mathcal{Y}_i))^{-1} \quad (3)$$

And for adjacent $\mathcal{T}_i$, $\mathcal{T}_j$, the pairwise potential is

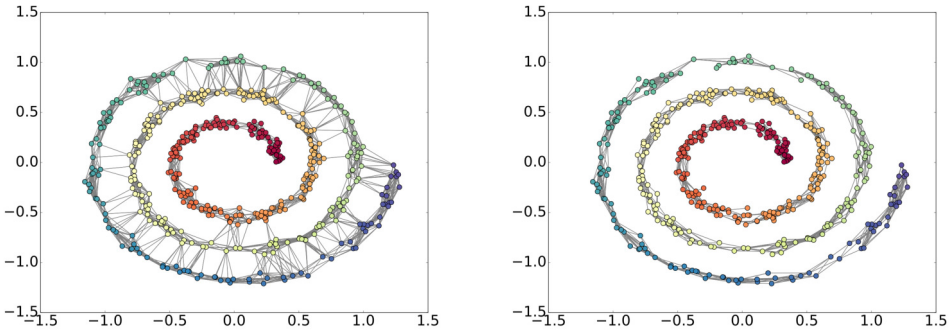
$$\psi(\mathcal{T}_i, \mathcal{T}_j) = (1 + \Gamma(\mathcal{T}_i, \mathcal{T}_j))^{-1} \quad (4)$$

D. Graph Denoising

TSBP uses the PMPNBP algorithm [19] to estimate the tangent space at every point. Once it has converged, we take the maximum likelihood estimate (MLE) of each marginal distribution $\mathcal{T}_i$ for each point $\hat{x}_i$. At the edge pruning stage, we remove any edges which disagree with the tangent space estimates (and presumably cross the manifold). Let α be a preselected pruning threshold. For each adjacent pair of points $\hat{x}_i, \hat{x}_j$ with tangent space estimates $\mathcal{T}_i$, $\mathcal{T}_j$, we define $e_{ij} = \hat{x}_j - \hat{x}_i$, and find angles ω_i and ω_j between e_{ij} and $\mathcal{T}_i$, and e_{ij} and $\mathcal{T}_j$, respectively. If $\min\{\omega_i, \omega_j\} > \alpha$, we remove the edge from the graph structure. After edge pruning, the graph may become disconnected. We iteratively add the shortest edge between two components until the graph is connected, using Kruskal's algorithm [29]. The resulting denoised graph is then used by a manifold learning algorithm to compute the embedding.

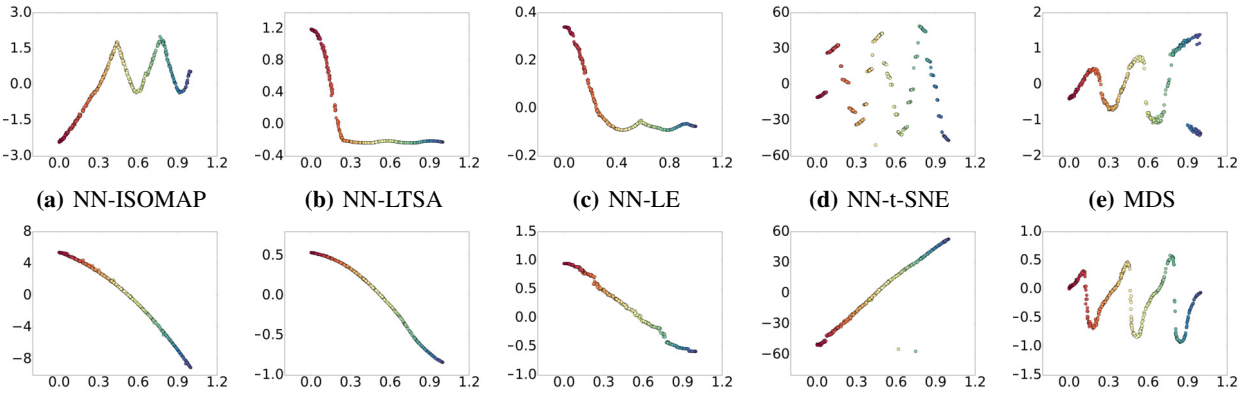
IV. Experiments

In our experiments, we demonstrate the effectiveness of our TSBP, in conjunction with several manifold learning algorithms, on sparse and noisy synthetic datasets. We use a simulated robot mapping task to emphasize TSBP's ability to preserve the local data topology. Finally, we apply TSBP to high-dimensional tactile sensory data



(a) k-Nearest Neighbors ($k = 12$) **(b)** TSBP

Fig. 3: A comparison of the neighborhood graphs from nearest-neighbors and TSBP on a 2D spiral dataset. All false edges crossing the manifold are pruned by our denoising algorithm. The parameterization of the underlying manifold is indicated by the color of each point, from red to blue.



(a) NN-ISOMAP **(b)** NN-LTSA **(c)** NN-LE **(d)** NN-t-SNE **(e)** MDS

(f) TSBP-ISOMAP **(g)** TSBP-LTSA **(h)** TSBP-LE **(i)** TSBP-t-SNE **(j)** Autoencoder

Fig. 4: A comparison of the embeddings of the 2D spiral data (see Figure 3) produced by various manifold learning algorithms, using nearest neighbors and our TSBP. In each graph, we plot the embedded coordinate (as obtained from the algorithm) versus the original parameter value that generated the data. With nearest neighbors, ISOMAP (a), LTSA (b), Laplacian Eigenmaps (c), and t-SNE (d) map distinct regions of the manifold to the same swaths of embedded coordinates. TSBP enables them all to produce accurate embeddings of the entire dataset.

to solve an object classification task.[1] For brevity in the figures, we refer to nearest-neighbors as "NN" and Laplacian Eigenmaps as "LE".

A. Embedding Synthetic Datasets

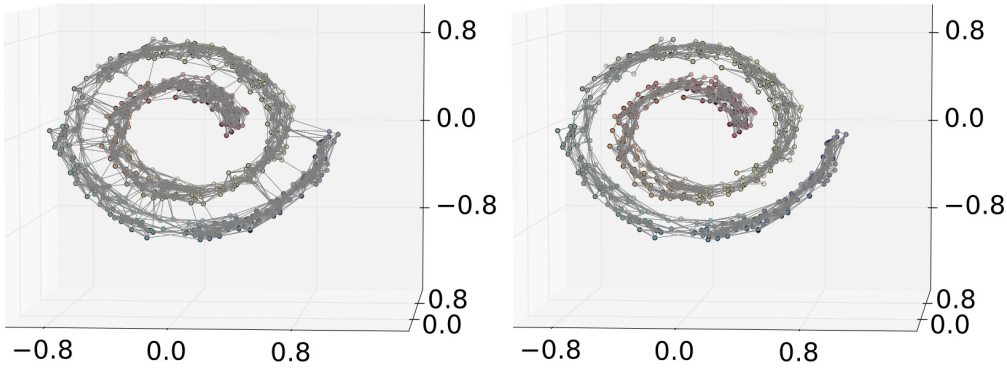
Consider the common example in manifold learning of sparse data uniformly sampled from a 2D spiral curve, with Gaussian observation noise. The ideal embedding for such a dataset is a straight line, as this implies a continuous, one-to-one relationship between the original parameter and the embedded coordinate. In Figure 3, we show the neighbor graph (a) initially constructed through k-NN and (b) after denoising based on TSBP. Initially, there are many false edges crossing the manifold; TSBP successfully prunes them all. In Figure 4, we learn the embedding using various manifold learning algorithms, with and without TSBP. The denoised neighborhood graph from TSBP allows these algorithms to compute an accurate embedding of the underlying manifold.

Consider data uniformly sampled from a 3D "Swiss Roll" manifold (another common example in manifold learning), with Gaussian observation noise. The ideal embedding for such a dataset is a flat rectangle. Like the previous experiment, TSBP successfully denoised the neighborhood graph by removing the false edges that were initially established by a nearest-neighbors search, as shown in Figure 5. As a result, we can see in Figure 6 that ISOMAP is able to use the denoised neighborhood graph to fully reconstruct the ideal embedding. Other manifold learning algorithms discard the width of the manifold, but preserve relative position along its length.

B. Mapping from Unstructured Sensor Observations

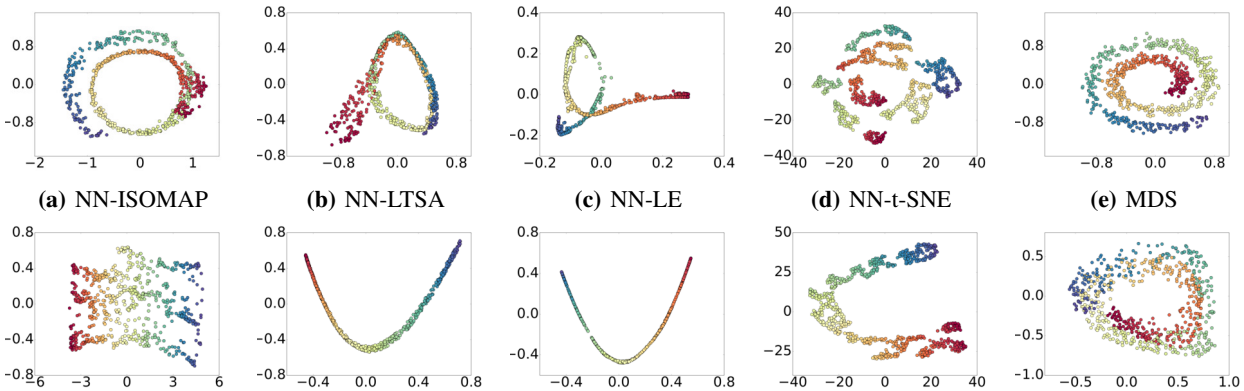
In the case of building a map of an environment from unstructured sensor data, discovering the relation between points can be very challenging [30]. To handle unordered or ambiguous data, it is important to accurately associate

[1]The authors extend considerable gratitude to Reviewer 2 for suggesting experiments with tactile data, which was a substantially meaningful contribution.



(a) k-Nearest Neighbors ($k = 9$) **(b)** TSBP

Fig. 5: A comparison of the neighborhood graphs from a nearest-neighbors search and from TSBP on a 3D swiss roll dataset. All false edges from the nearest-neighbors are pruned by our denoising algorithm. Note that the color profile is identical to the spiral example (Figure 3).



(a) NN-ISOMAP **(b)** NN-LTSA **(c)** NN-LE **(d)** NN-t-SNE **(e)** MDS

(f) TSBP-ISOMAP **(g)** TSBP-LTSA **(h)** TSBP-LE **(i)** TSBP-t-SNE **(j)** Autoencoder

Fig. 6: A comparison of the embeddings of the 3D swiss roll data (see Figure 5) produced by various manifold learning algorithms, using nearest neighbors and our TSBP. In each graph, we plot the 2D embedded coordinates, and look for an unrolled, flat Swiss Roll. With nearest neighbors, ISOMAP (a), LTSA (b), Laplacian Eigenmaps (c), and t-SNE (d) all fail to unroll the manifold. TSBP enables ISOMAP (f) to produce an accurate embedding of the entire dataset. With TSBP, LTSA (g), Laplacian Eigenmaps (h), and t-SNE (i) produce improved embeddings that don't map any distinct regions to the same swaths of coordinates, but do discard the width of the manifold.

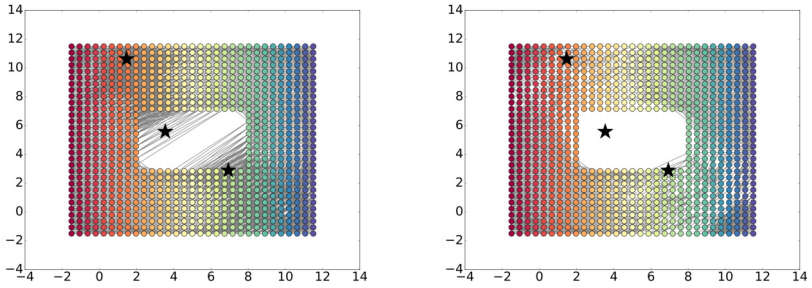
sensor observations that are taken at similar locations as neighbors. This problem of accurate neighbor association between unordered and ambiguous observation data can be formulated as the problem of constructing the accurate neighborhood graph among these observation data.

To demonstrate the effectiveness of our TSBP in dealing with unordered and ambiguous observations, we present an experiment where unstructured sensor readings of a simulated environment are used to build a map of the area. These sensor readings consist only of the distance to three beacons scattered across the environment. (Consider, for example, cell phone towers and bluetooth beacon trackers.) The true locations of the sensor readings and the layout of the beacons are illustrated in Figure 7. The combination of sparse 3D measurements and poor beacon placement causes points which are distant in real space to appear very close in sensor space, preventing existing manifold learning algorithms from accurately reproducing the map. In Figure 7, we compare the neighborhood graph from k-nearest neighbors and TSBP, and in Figure 8, we learn the embedding using ISOMAP and t-SNE, with and without TSBP. By removing the false edges, we're able to better preserve the local topology of the data, and produce a more accurate map.

C. *Classification of High-Dimensional Tactile Data*

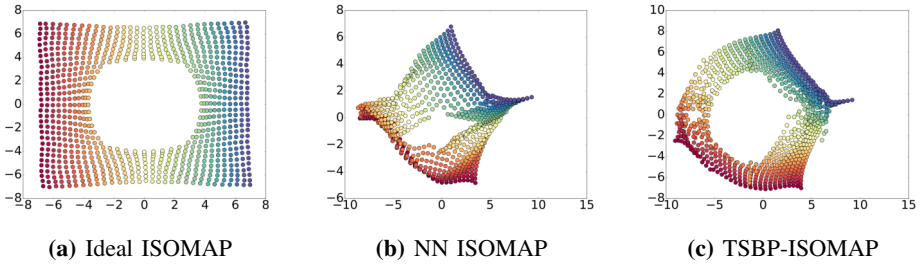
Humans are able to learn a great deal of information about an object via touch. Hence, tactile sensing is an area of great interest in robotics, especially in the context of grasping and manipulation [31], [32], [33], [34]. In order to design future robot grippers, it will be important to understand the human grasp in great detail. Sundaram *et al.* [31] developed a scalable tactile glove (STAG) to gather data about the human grasp. As shown in Figure 9, the glove is embedded with 548 force sensors, each able to measure from 30 to 500 mN, so the data obtained from the STAG is detailed, but also high dimensional.

Sparsity grows exponentially with the number of



(a) k-Nearest Neighbors ($k = 15$) **(b)** TSBP

Fig. 7: A comparison of the neighborhood graphs from nearest-neighbors and TSBP on the beacon dataset, projected onto the original map. Almost all false edges from the nearest neighbors are pruned by our denoising algorithm. The beacon locations are overlaid as black stars.



(a) Ideal ISOMAP **(b)** NN ISOMAP **(c)** TSBP-ISOMAP

(d) Ideal t-SNE **(e)** t-SNE **(f)** TSBP-t-SNE

Fig. 8: A comparison of the 2D embeddings of beacon distance data produced by TSBP with various other manifold learning algorithms. We compare ISOMAP and t-SNE embeddings, each with different neighborhood graphs. For ISOMAP, in (a), we have an ideal graph with no false edges; in (b), we have the k-nearest neighbors graph; in (c), we have the denoised graph from TSBP. For t-SNE, in (d), we use distance along an ideal neighborhood graph with no false edges; in (e), we have standard t-SNE; in (f), we use distance along the denoised graph from TSBP. With nearest-neighbors, ISOMAP overlaps several local neighborhoods in its corners, and t-SNE fails to discover the latent mapping. Only with our TSBP can these algorithms produce accurate embeddings that preserve both the global and local topology.

dimensions, so large amounts of data are required to accurately represent a given high-dimensional feature space. For classification of grasped objects, Sundaram *et al.* gathered 135,000 frames, and achieved high accuracy with a deep convolutional neural network (CNN) [31]. But extensive data collection is not always feasible; in these cases, dimensionality reduction may be necessary to meaningfully interpret the data.

To demonstrate TSBP's ability to meaningfully interpret high-dimensional tactile data, we use it as part of an object classification task. We use several dimensionality reduction algorithms to embed a small subset of the STAG tactile dataset into a low-dimensional space, and classify the 26 objects (and empty hand) using a Gaussian process classifier (GPC) with an RBF kernel. Dimensionality reduction is necessary, as the RBF kernel relies on the Euclidean metric [35], which performs poorly in high dimensional spaces [36].

Specifically, we use PCA to embed the data into 200 dimensions, and then use t-SNE, nearest-neighbors t-SNE, and TSBP t-SNE to embed the data into 10 dimensions; we also use an autoencoder to directly embed the original data to 10 dimensions. We then train a GPC on each embedding. As a baseline, we train a GPC without any dimensionality reduction, and also compare to the deep CNN presented by Sundaram *et al.* [31].

To ensure reliability of our results, we perform this experiment on 100 different random subsets of the STAG dataset. Each dataset has 24 samples per class, with a 50/50 train-test split. In Table I, we indicate the average top-1 and top-3 accuracy. TSBP's minimum improvement over nearest-neighbors was 6%, and the maximum was

TABLE I: Top-1 and Top-3 accuracy for the GPC on tactile data, organized by dimensionality reduction method. We compare t-SNE, NN t-SNE, TSBP t-SNE, and an Autoencoder. We also compare to a GPC trained on the original data (with no dimensionality reduction), a deep CNN, and the expected accuracy of making random predictions. TSBP t-SNE clearly has the best accuracy.

	Random Chance	No Dimensionality Reduction	Autoencoder	t-SNE	NN t-SNE	TSBP t-SNE	Deep CNN
Top-1 Accuracy	0.0370	0.0402	0.0612	0.0421	0.0380	**0.1874**	0.0897
Top-3 Accuracy	0.1111	0.1140	0.1551	0.1156	0.1120	**0.2972**	0.1888

Fig. 9: The STAG glove, created by Sundaram *et al.* [31].

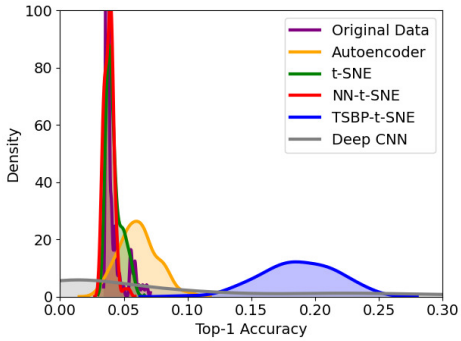


Fig. 10: Kernel density estimation of the accuracy for the tactile data classification task, organized by method. This experiment used a dataset size of 648.

21%. In Figure 10, we show a kernel density plot of the accuracies, which clearly demonstrates the consistent improvement of TSBP over nearest neighbors. Our TSBP clearly produces a more reliable neighborhood graph than nearest-neighbors, leading to a better low-dimensional embedding. Subsequently, we achieve higher accuracy when training on small amounts of data.

Toward reproducibiliy, we provide the hyperparameters used in the experiments. In every experiment, ISOMAP, LTSA, Laplacian Eigenmaps, and TSBP share a neighborhood size parameter k. TSBP has an edge pruning threshold α, and t-SNE has a perplexity parameter. We list hyperparameter values for the experiments in Table II.

Our autoencoder uses 3 fully-connected layers for encoding and for decoding. The latent variable layer has the same number of neurons as the manifold's dimension. For the encoding network, the first layer has 64 neurons, and the second and third layers both have 32 neurons. For the decoding network, the first and second layers both have 32 neurons, and the third layer has 64 neurons. The RELU activation function is used throughout, except for the latent variable layer, which uses $\tanh$.

TABLE II: Hyperparameter values for the experiments.

	2D Spiral	Swiss Roll	Beacon Mapping	Tactile Data
k	12	9	15	12
α	$\cos(\pi/3)$	$\cos(\pi/3)$	0.98	$\cos(\pi/3)$
Perplexity	30	30	30	30

V. Conclusions

We have presented TSBP as a new method of discovering the geometric structure of data for the purposes of manifold learning. TSBP has demonstrated that belief propagation allows us to robustly estimate the tangent space, even in the presence of sparse and noisy data. This allows us to reliably remove false edges obtained by a nearest neighbors search, thereby enabling manifold learning algorithms to more accurately embed the data. The primary drawback of TSBP is runtime – belief propagation is computationally expensive on large graphical networks. The time complexity is linear with respect to number of points and quadratic with respect to dimension, and PMPNBP enables linear-time computation of message products with respect to the number of particles [19].

We have not considered the impact of outliers; preliminary investigations have shown that they skew the initial tangent space estimates, but TSBP is still able to make an accurate estimate. It may be beneficial to use L1-PCA for initial tangent space estimates, but precise computation is intractable for high-dimensional datasets [37].

References

[1] G. V. Trunk, "A problem of dimensionality: A simple example," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 3, pp. 306–307, 1979.

[2] N. Roy and G. J. Gordon, "Exponential family pca for belief compression in pomdps," in *Advances in Neural Information Processing Systems*, 2003, pp. 1667–1674.

[3] M. T. Ciocarlie and P. K. Allen, "Hand posture subspaces for dexterous robotic grasping," *The International Journal of Robotics Research*, vol. 28, no. 7, pp. 851–867, 2009.

[4] J. Morimoto, S.-H. Hyon, C. G. Atkeson, and G. Cheng, "Low-dimensional feature extraction for humanoid locomotion using kernel dimension reduction," in *2008 IEEE International Conference on Robotics and Automation*. IEEE, 2008, pp. 2711–2716.

[5] S. Remy and A. M. Howard, "In situ interactive teaching of trustworthy robotic assistants," in *2007 IEEE International Conference on Systems, Man and Cybernetics*. IEEE, 2007, pp. 1280–1285.

[6] A. Angelova, L. H. Matthies, D. M. Helmick, and P. Perona, "Dimensionality reduction using automatic supervision for vision-based terrain learning," in *Robotics: Science and Systems III, June 27-30, 2007, Georgia Institute of Technology, Atlanta, Georgia, USA*, 2007. [Online]. Available: http://www.roboticsproceedings.org/rss03/p29.html

[7] M. Gualtieri, A. Ten Pas, K. Saenko, and R. Platt, "High precision grasp pose detection in dense clutter," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 598–605.

[8] N. Sünderhauf, O. Brock, W. Scheirer, R. Hadsell, D. Fox, J. Leitner, B. Upcroft, P. Abbeel, W. Burgard, M. Milford *et al.*, "The limits and potentials of deep learning for robotics," *The International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 405–420, 2018.

[9] O. C. Jenkins and M. J. Matarić, "A spatio-temporal extension to isomap nonlinear dimension reduction," in *Proceedings of the Twenty-First International Conference on Machine Learning*, ser. ICML '04. New York, NY, USA: Association for Computing Machinery, 2004, p. 56. [Online]. Available: https://doi.org/10.1145/1015330.1015357

[10] J. Romero, T. Feix, C. H. Ek, H. Kjellström, and D. Kragic, "Extracting postural synergies for robotic grasping," *IEEE Transactions on Robotics*, vol. 29, no. 6, pp. 1342–1352, 2013.

[11] C.-C. Ho, K. F. MacDorman, and Z. D. Pramono, "Human emotion and the uncanny valley: a glm, mds, and isomap analysis of robot video ratings," in *2008 3rd ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2008, pp. 169–176.

[12] J. Sinapov and A. Stoytchev, "Detecting the functional similarities between tools using a hierarchical representation of outcomes," in *2008 7th IEEE International Conference on Development and Learning*. IEEE, 2008, pp. 91–96.

[13] J. Sturm, V. Pradeep, C. Stachniss, C. Plagemann, K. Konolige, and W. Burgard, "Learning kinematic models for articulated objects," in *Twenty-First International Joint Conference on Artificial Intelligence*, 2009.

[14] J. B. Tenenbaum, V. De Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[15] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[16] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[17] A. Tsoli and O. C. Jenkins, "Neighborhood denoising for learning high-dimensional grasping manifolds," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2008, pp. 3680–3685.

[18] K. Pearson, "Liii. on lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, no. 11, pp. 559–572, 1901.

[19] K. Desingh, S. Lu, A. Opipari, and O. C. Jenkins, "Efficient nonparametric belief propagation for pose estimation and manipulation of articulated objects," *Science Robotics*, vol. 4, no. 30, 2019. [Online]. Available: https://robotics.sciencemag.org/content/4/30/eaaw4523

[20] W. S. Torgerson, "Theory and methods of scaling." 1958.

[21] Z. Zhang and H. Zha, "Nonlinear dimension reduction via local tangent space alignment," in *International Conference on Intelligent Data Engineering and Automated Learning*. Springer, 2003, pp. 477–481.

[22] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[23] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Wiley-Interscience, 2000.

[24] J. Park, Z. Zhang, H. Zha, and R. Kasturi, "Local smoothing for manifold learning," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, vol. 2. IEEE, 2004, pp. II–II.

[25] R. S. Zemel and M. Á. Carreira-Perpiñán, "Proximity graphs for clustering and manifold learning," in *Advances in neural information processing systems*, 2005, pp. 225–232.

[26] H. Chen, G. Jiang, and K. Yoshihira, "Robust nonlinear dimensionality reduction for manifold learning," in *18th International Conference on Pattern Recognition (ICPR'06)*, vol. 2. IEEE, 2006, pp. 447–450.

[27] X. Xing, S. Du, and K. Wang, "Robust hessian locally linear embedding techniques for high-dimensional data," *Algorithms*, vol. 9, no. 2, p. 36, 2016.

[28] J. M. Lee, "Smooth manifolds," in *Introduction to Smooth Manifolds*. Springer, 2013, pp. 1–31.

[29] J. B. Kruskal, "On the shortest spanning subtree of a graph and the traveling salesman problem," *Proceedings of the American Mathematical society*, vol. 7, no. 1, pp. 48–50, 1956.

[30] S. Agarwal, Y. Furukawa, N. Snavely, I. Simon, B. Curless, S. M. Seitz, and R. Szeliski, "Building rome in a day," *Commun. ACM*, vol. 54, no. 10, p. 105–112, Oct. 2011. [Online]. Available: https://doi.org/10.1145/2001269.2001293

[31] S. Sundaram, P. Kellnhofer, Y. Li, J.-Y. Zhu, A. Torralba, and W. Matusik, "Learning the signatures of the human grasp using a scalable tactile glove," *Nature*, vol. 569, no. 7758, pp. 698–702, 2019.

[32] W. Yuan, S. Dong, and E. H. Adelson, "Gelsight: High-resolution robot tactile sensors for estimating geometry and force," *Sensors*, vol. 17, no. 12, p. 2762, 2017.

[33] Z. Kappassov, J.-A. Corrales, and V. Perdereau, "Tactile sensing in dexterous robot hands," *Robotics and Autonomous Systems*, vol. 74, pp. 195–220, 2015.

[34] Y. Chebotar, O. Kroemer, and J. Peters, "Learning robot tactile sensing for object manipulation," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 3368–3375.

[35] C. R. Souza, "Kernel functions for machine learning applications," 2010. [Online]. Available: http://crsouza.com/2010/03/17/kernel-functions-for-machine-learning-applications/

[36] C. C. Aggarwal, A. Hinneburg, and D. A. Keim, "On the surprising behavior of distance metrics in high dimensional space," in *International conference on database theory*. Springer, 2001, pp. 420–434.

[37] P. P. Markopoulos, G. N. Karystinos, and D. A. Pados, "Some options for l1-subspace signal processing," in *ISWCS 2013; The Tenth International Symposium on Wireless Communication Systems*. VDE, 2013, pp. 1–5.