

Estimating Manifold Dimension by Inversion Error

Shawn Martin

Sandia National Laboratories
Department of Computational Biology, 9212
PO Box 5800, MS 0310
Albuquerque, NM 87110
smartin@sandia.gov

Alex Bäcker

Sandia National Laboratories &
California Institute of Technology
Department of Computational Biology (Sandia)
Biology Division (Caltech)
Beckman Institute, MC 139-74
Pasadena, California 91125
alex@caltech.edu

ABSTRACT

Video and image datasets can often be described by a small number of parameters, even though each image usually consists of hundreds or thousands of pixels. This observation is often exploited in computer vision and pattern recognition by the application of dimensionality reduction techniques. In particular, there has been recent interest in the application of a class of nonlinear dimensionality reduction algorithms which assume that an image dataset has been sampled from a manifold.

From this assumption, it follows that estimating the dimension of the manifold is the first step in analyzing an image dataset. Typically, this estimate is obtained either by using *a priori* knowledge, or by applying one of the various statistical and geometrical methods available. Once an estimate is obtained, it is used as a parameter for the nonlinear dimensionality reduction algorithm.

In this paper, we consider reversing this approach. Instead of estimating the dimension of the manifold in order to obtain a low dimensional representation, we consider producing low dimensional representations in order to estimate the dimensionality of the manifold. By varying the dimensionality parameter, we obtain different low dimensional representations of the original dataset. The dimension of the best representation should then correspond to the actual dimension of the manifold.

In order to determine the best representation, we propose a metric based on inversion. In particular, we propose that a good representation should be invertible, in that we should be able to reverse the reduction algorithm's transformation to obtain the original dataset. By coupling this metric with *any* reduction algorithm, we can estimate the dimensionality of an image manifold. We apply our method in the context of locally linear embedding (LLE) and Isomap to six frequently used examples and two image datasets.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SAC '05, March 13-17, 2005, Santa Fe, New Mexico, USA.
Copyright 2005 ACM 1-58113-964-0/05/0003 ...\$5.00.

1. INTRODUCTION

Image and video datasets often contain scenes where only a few things change. Thus, despite the fact that the datasets themselves are very large and quite high dimensional, they can often be described by only a few parameters [3], [13], [25]. In fact, it is typically assumed that such datasets are actually low dimensional manifolds embedded in a high dimensional space [14], [22].

It follows that the estimation of the dimensionality of such manifolds is an important first step in their analysis. As such, various algorithms for this purpose have been developed. These algorithms have been developed in the context of statistics [8], [1], [15], [26] and geometry [5], [12], [6], [10]. The statistical approaches often focus on the use of variance in the dataset to determine intrinsic dimensionality, while the geometric approaches often use topological considerations based on the scaling of various quantities with neighborhood size.

The next step in the analysis of an image manifold is obtaining a low dimensional representation of the original dataset. Recently, a number of nonlinear algorithms have been developed to provide such representations. These algorithms include kernel principal component analysis [21], locally linear embedding (LLE) [19], [20], Isomap [24], [16] Hessian LLE [7], Lapacian eigenmaps [2], semidefinite embedding [28], and manifold charting [4], among others. Typically, these algorithms require as input a parameter specifying the dimension of the resulting low dimensional representation. Although some of the reduction algorithms provide their own estimates of the dimension, it is common to provide this parameter using one of the aforementioned estimation techniques.

In this paper, we consider reversing this standard approach. We consider analyzing the results of the dimensionality algorithms in order to obtain an estimate of the dimension of the manifold. In particular, we vary the dimensionality estimate parameter, and examine the resulting low dimensional representations. These representations are ranked according to a quality measure based on inversion. In theory, the dimension of the best representation will correspond to the actual dimension of the manifold.

Our technique is based on the idea that a dimensionality reduction algorithm should preserve information on a global scale, as measured using bijection. From this point of view our method would be considered a geometric method, albeit

global, in contrast to the geometric methods cited previously (all local). In addition, it is unique in that it is designed around the results of *any* available dimensionality reduction algorithm. It both estimates image manifold dimension and validates dimensionality reduction algorithms.

Our paper is organized as follows: in Section 2 we introduce our quality measure, which is based on the inversion of the reduced dimensional manifold; in Section 3 we provide background on LLE and Isomap; in Section 4 we apply our method in the context of LLE and Isomap; and in Section 5 we discuss our method and possible improvements.

2. INVERSION

Given an input dataset $X = \{X_1, \dots, X_N\} \subset \mathbb{R}^D$, dimensionality reduction algorithms such as LLE and Isomap provide a reduced dimensional representation $Y = \{Y_1, \dots, Y_N\} \subset \mathbb{R}^d$ of the original dataset X . The underlying assumption behind these algorithms is that X lies on a manifold \mathcal{M} embedded in \mathbb{R}^D with intrinsic dimensionality d^* . The estimated dimension d of d^* is provided by the user as a parameter to the algorithm.

In addition to the reduced dataset Y , these algorithms implicitly provide a map $f : X \rightarrow Y$. Although this map is constructed to have any number of desirable properties, such as local linearity (LLE) or geodesic distance preservation (Isomap), it should from first principles also satisfy more elementary properties, such as continuity, invertibility, possibly distance preservation, et cetera.

We focus on the simplest of these properties: invertibility. In particular, we expect that if Y is a good low dimensional representation of X , then we should be able to go back and forth from X to Y with no loss of information. Mathematically, we expect that an inverse function $f^{-1} : Y \rightarrow X$ exists such that $f^{-1}(f(X_i)) = X_i$ for all i . Furthermore, such an inverse should exist only if the estimated dimension d is greater than or equal to the intrinsic dimension d^* . We conclude that the existence of f^{-1} provides a way to assess the intrinsic dimension d^* of the manifold.

To determine the existence of f^{-1} , we consider the squared residual error $r_d = \sum_i \|f_d^{-1}(f_d(X_i)) - X_i\|^2$, where $f_d : X \rightarrow Y$ is map produced by the dimensionality reduction algorithm which depends on the estimate d of the intrinsic dimensionality, and $f_d^{-1} : Y \rightarrow X$ is a proposed inverse to f_d . By examining the behavior of r_d , which we call the *inversion error*, we can determine the existence of f_d^{-1} and then estimate the actual dimension d^* of the manifold \mathcal{M} .

Obviously, we want to minimize r_d over different values of d . In theory, r_d should be zero when $d \geq d^*$ so that we may determine d^* by finding the smallest value of d such that $r_d = 0$. In practice, however, r_d is seldom actually zero, so that we determine d^* by minimizing both d and r_d simultaneously. Although this notion could be formalized further (by minimizing, for example, $r_d + d$), we are at present more interested in the general behavior of r_d than a precisely formulated optimization problem.

3. LLE & ISOMAP

We first consider our method in the context of LLE [19], [20]. As mentioned previously, LLE is a method for nonlinear dimensionality reduction which takes as input a dataset $X = \{X_1, \dots, X_N\} \subset \mathbb{R}^D$, assumed to lie on a manifold \mathcal{M} with dimension d^* . The user of LLE must supply an es-

timate d of d^* , with which LLE constructs a dataset $Y = \{Y_1, \dots, Y_N\}$ in \mathbb{R}^d . LLE is most often illustrated by unrolling a two dimensional spiral, known as the Swiss roll, into a rectangle. This and other examples will be examined later and can be found in [19], [20].

Although the details of LLE can be found in [20], we provide a brief description in order to fully explain our method. The first step in LLE is to solve for the location of each point $X_i \in \mathbb{R}^D$ in terms of its K nearest neighbors. This step is performed simultaneously for every point $X_i \in X$ by solving

$$\begin{aligned} \min \quad & E(W) = \sum_i \|X_i - \sum_j W_{ij} X_j\|^2 \\ \text{subject to} \quad & \begin{cases} W_{ij} = 0 \text{ if } X_i \text{ not neighbor } X_j \\ \sum_j W_{ij} = 1 \text{ for every } i. \end{cases} \end{aligned} \quad (1)$$

This problem has a closed form solution and assures not only that each approximation $X_i \approx \sum_j W_{ij} X_j$ lies in the subspace spanned by the K neighbors of X_i , but also that the solution W is translationally invariant.

Once the reconstruction weights W have been obtained, the next step in LLE is to perform a quadratic minimization. This minimization is used to construct Y so that it is locally similar to X . The minimization is in practice solved via an eigenvalue problem, and the embedding is provided by the first d eigenvectors (actually the second through $(d+1)$ st eigenvectors), corresponding to the smallest eigenvalues.

The estimate d of the intrinsic dimension d^* of the manifold must be provided by the user of LLE. If d is an underestimate of d^* , we suffer a loss of information, and if d is an overestimate of d^* , then LLE will include arbitrary dimensions. These observations have been discussed in [17] and in [20].

The method of [17], which we use as a benchmark comparison for our method, estimates the dimension of \mathcal{M} by examining the eigenvalues of the matrix used to solve the quadratic minimization in the calculation of the LLE embedding. The work in [17] shows that these eigenvalues should be identically zero until the correct dimension is achieved, so that we can estimate the dimension of \mathcal{M} by counting the number of zero eigenvalues. Unfortunately, however, the work in [20] shows that these eigenvalues are largely uninformative unless using very well behaved data sets.

In order to implement our method in the context of LLE, and to compare our method with the method in [17], we must provide f_d and f_d^{-1} . Fortunately, candidates for these maps are provided by [20]. In particular, we can use the maps defined by $f_d(U) = \sum_j w_j Y_j$, where the w_j are the reconstruction weights calculated from the K nearest neighbors of U according to (1). Similarly, $f_d^{-1}(V) = \sum_j w_j X_j$, where now the w_j are computed according to (1), but using $\{Y_1, \dots, Y_N\}$ in place of $\{X_1, \dots, X_N\}$.

Since the application of our method to LLE uses maps f_d and f_d^{-1} that are specific to LLE, we also wanted to make sure that our method would generalize to algorithms besides LLE. We therefore applied our method to Isomap. Isomap is another nonlinear dimensionality reduction algorithm which takes as input a dataset $X = \{X_1, \dots, X_N\}$ along with an estimate d of the intrinsic dimension of X . Isomap then produces a reduced dimensional representation Y by preserving corresponding geodesic distances in X . We do not give details except to say that Isomap first computes a graph based representation of the geodesic distances in X and then uses multi-dimensional scaling to produce Y . As an added ben-

efit, the residual variance computed by multi-dimensional scaling can be used to estimate the dimensionality of the manifold in question See [24] for details.

For our purposes, it is enough to know that Isomap produces a map $f_d : X \rightarrow Y$ for each estimate d of the dimension of X . In this case, we are taking f_d to be the actual correspondence between X and Y given by Isomap, where X and d are Isomap’s inputs. We need only to produce an additional function f_d^{-1} so that we may compute the inversion error r_d . Although this map could be obtained using any number of methods, for this example we use Support Vector Regression [23].

Support Vector Regression (SVR) is a technique for nonlinear regression (necessary since we are using a nonlinear reduction algorithm), which fits curves to data based on the use of *kernel functions* and an ϵ -tolerance for errors. As with Isomap, we do not describe SVR in much detail. We say only that SVR is based on a quadratic programming problem which uses the kernel functions to introduce nonlinearity. The user of SVR must provide a kernel function with parameters appropriate to the dataset at hand as well as an ϵ -tolerance. In the following examples (except for the lattice) we used a Gaussian kernel $k(Y_i, Y_j) = \exp(-\gamma\|Y_i - Y_j\|^2)$ with $\gamma = .5$, and an ϵ -tolerance of .1. For the lattice we used a linear kernel. In all cases, we used the *SVM^{light}* program [11] to train our SVRs.

4. EXAMPLES

We first tested our algorithm using LLE and compared our results with the method in [17]. We begin with six simple two-dimensional manifolds: a lattice, the Swiss roll, the S-curve, the twin peaks, the punctured sphere, and a sphere. The lattice consists of a 10x11 grid embedded in the plane $x + y + z = 0$, the Swiss roll, S-curve, twin peaks and punctured sphere were obtained from [19], [20], and the sphere is a randomly sampled unit sphere. These manifolds are shown in Figure 1(a).

For each of these manifolds, we computed both the LLE ($K=12$) eigenvalues and the inversion error. With these quantities, we compared our method with the method in [17]. To facilitate this comparison, we plot normalized versions of these quantities vs. dimension, where the normalization is performed by dividing each quantity by either the largest eigenvalue or the largest inversion error. The results are shown in Figure 1(b). These plots confirm the observations made in [20] about the spurious practical relation between the number of zero eigenvalues and the intrinsic dimension of the manifold. The inversion error, on the other hand, seems to perform well on every example. In particular, the error is high until the correct dimension is obtained, at which point the error changes to near zero. (In the case of the sphere we note that the actual dimension of the manifold is two, although a sphere requires a three dimensional embedding.)

We also tested our algorithm (again using LLE) on two image datasets. The first dataset consists of images of B. Frey’s face in various poses and was used in [19], [20] as an example. We obtained the dataset from S. Roweis’ website [18]. The dataset consists of 1965 grayscale images, each containing 20×28 pixels. We scaled the face dataset so that each pixel value was between -1 and 1. The second dataset consists of images of handwritten “2”s and was obtained from K. Weinberger’s website [27] in the context of

semidefinite embedding [28]. However, the original data was obtained from the USPS data set of handwritten digits [9], and we use it here because a similar dataset was used as an example in [24]. The dataset consists of 638 grayscale images, each containing 16×16 pixels.

The results of our algorithm applied to the face and twos datasets are shown in Figure 2. In the case of the faces data, we used $K = 11$ and predicted an intrinsic dimension of 2 or 3. This is consistent with the results in [19], in which the dimension is thought to be 2, as well as the results in [4], where the dimension is thought to be 3. To provide a further check of these predictions, we also produced visualizations using Isomap (discussed next). In each of these visualizations we used four rows of images, where the rows correspond to the first through fourth (from top to bottom) components produced by Isomap. In each row the images were again ordered, this time by the value of the projection in the corresponding component. In the case of the face dataset, we see that the first component corresponds to pose (left to right), and the second component corresponds to facial expression (smile to frown). It is more difficult to interpret the third and fourth rows, in agreement with our predictions.

When considering the twos dataset, we used $K = 15$ and predicted an intrinsic dimension of 5. This prediction is almost certainly too high, as our prediction using Isomap (discussed next) was 3. In addition, looking at the Isomap projections, we can see that the first component corresponds to image height and width (from short and wide to tall and narrow), while the second component corresponds to a curly tail (from no tail to curly tail). Again the third and fourth components are more difficult to interpret, although the third component does exhibit a change in aspect (the digit changes from pointing down and left to up and left). In any case, ambiguity concerning the dimension of this dataset was also found in [24], [28].

We next tested our dimensionality estimation technique using Isomap and SVR on the same examples that we used to test the LLE version of our method (we used the same values for K that we used for LLE). The results of these tests are shown in Figures 1(c) and 2. In Figure 1(c), we show both our inversion error and the residual variance measure provided by Isomap. The residual variance measure works in the same way that our measure works, in that we can estimate the dimensionality of the input by locating the dimension at which there is no longer any residual variance. As can be seen in Figure 1(c), both the inversion error and the residual variance identify correctly the dimension in each of the examples.

In Figure 2, we see in the case of the face dataset that our algorithm gives results similar to those obtained using Isomap’s residual variance. Namely, both methods predict an intrinsic dimension of 3. When considering the twos dataset, however, our method gives a relatively clear indication that the intrinsic dimension is 3, while the residual variance measure does not give a very clear prediction (possibly 4).

5. DISCUSSION

We have proposed a novel method for estimating the intrinsic dimensionality of an image manifold. Our method is based on an *a posteriori* analysis of the results of a dimensionality reduction algorithm. In particular, we con-

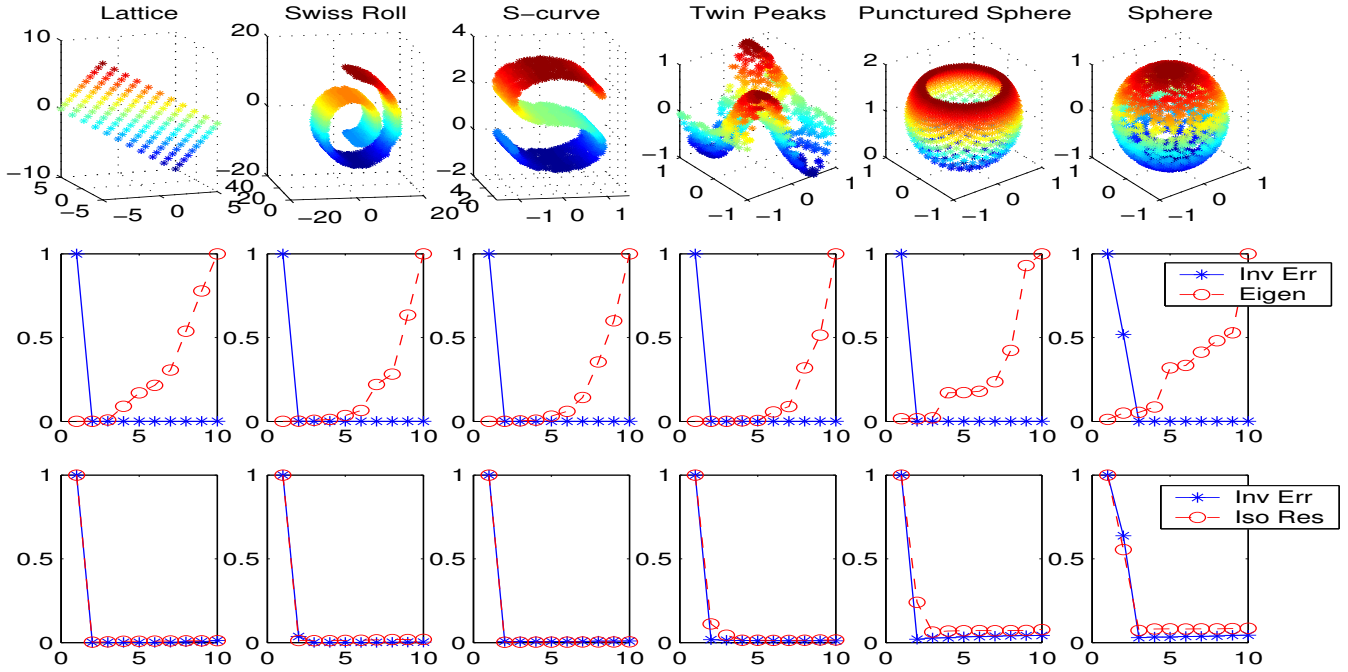


Figure 1: Six Examples. Here we show the results of our algorithm on six manifolds. In the first row (a) we show the manifolds; in the second row (b) we show a normalized version of the LLE inversion error, along with a normalized version of the first ten eigenvalues from LLE, both versus dimension; and in the third row (c) we show a normalized version of the Isomap inversion error, along with the Isomap residual error measure (also normalized).

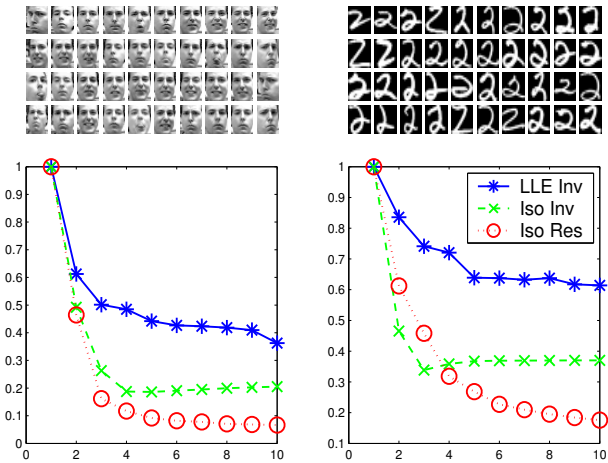


Figure 2: Image Manifolds. Here we show the results of our algorithm on two image manifolds. The plots on the left show the results of our algorithm on the faces data, and on the right for the two dataset. Above each plot we show images from the Isomap representation. The first row contains ordered images from the first Isomap component, the second row contains ordered images from the second Isomap component, et cetera. In the plots below the images we show the LLE inversion error, the Isomap SVR inversion error, and the Isomap residual variance, all versus dimension.

consider how readily a reduced dimensional representation of a manifold can be inverted. We have shown that the invertibility of a reduced dimensional manifold is related to the dimension of the original manifold, and that our method performs well in practice. In the case of LLE, our method performs better than the existing method, and in the case of Isomap, we obtain similar or better estimates of the intrinsic dimensionality, comparing against Isomap’s residual variance measure.

A particular strength of our method is its generality. Our method does not depend on the particular algorithm used to perform the dimensionality reduction, nor on the particular method used to discover the inverse mapping from the reduced dimensional representation to the original manifold. Of course, it is also true that our method depends on the success of both the dimensionality reduction algorithm and the method used to discover the inverse map. Indeed, we saw in Figure 2 that our estimate of the dimensionality changed from 5 to 3 for the two dataset when we used Isomap in place of LLE.

Although we have shown in principal that our method is useful in estimating the intrinsic dimensionality of an image manifold, there are additional questions which we have not addressed. First, what method is best for learning the inverse map $f_d^{-1} : Y \rightarrow X$? Here we should consider both the quality of the final map $f_d^{-1} : Y \rightarrow X$, as well as the speed of the algorithm used to learn the map. In our examples the LLE map was much faster than the SVR map, although it could be argued that the SVR map was more accurate, especially on the image datasets. Second, how can we automate our method? The behavior of r_d is sometimes

difficult to interpret, and does not provide an automatic determination of the dimension of the manifold. As suggested previously, it may be helpful to consider alternate quantities such as $r_d + d$ in place of r_d . Finally, and more generally, what other properties of a map produced by a dimensionality reduction algorithm can we consider in an effort to either estimate the dimensionality of a manifold or just generally validate the reduced dimensional representation?

6. ACKNOWLEDGEMENTS

This work was supported by the Department of Energy's Office of Advanced Scientific Computing Research (OASCR) Applied Mathematics program, specifically under Mathematics, Information and Computational Sciences (MICS) project 547 task 02.09. Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under Contract DE-AC04-94AL85000.

7. REFERENCES

- [1] D. Banks and R. Olszewski. Estimating local dimensionality. In *Proceedings of the Statistical Computing Section of the American Statistical Association*. ASA, 1997.
- [2] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396, 2003.
- [3] D. Beymer and T. Poggio. Image representation for visual learning. *Science*, 272:1905–1909, 1996.
- [4] M. Brand. Charting a manifold. In *Advances in Neural Information Processing Systems*, volume 15, pages 961–968. MIT Press, 2003.
- [5] J. Bruske and G. Summer. Intrinsic dimensionality estimation with optimally topology preserving maps. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(5):572–575, 1998.
- [6] F. Camastra and A. Vinciarelli. Estimating the intrinsic dimension of data with a fractal-based method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(10):1404–1407, 2002.
- [7] D. Donoho and C. Grimes. Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. *Proceedings of the National Academy of Science*, 100:5591–5596, 2003.
- [8] K. Fukunaga and D. Olsen. An algorithm for finding intrinsic dimensionality of data. *IEEE Transactions on Computers*, 20(2):176–183, 1971.
- [9] J. J. Hull. A database for handwritten text recognition research. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5):550–554, 1994.
- [10] D. Hundley and M. Kirby. Estimation of topological dimension. In *SIAM International Conference on Data Mining*, pages 194–202, 2003.
- [11] T. Joachims. Making large-scale svm learning practical. In *Advances in Kernel Methods – Support Vector Learning*. MIT Press, 1999.
- [12] B. Kegl. Intrinsic dimension estimation using packing numbers. In *Advances in Neural Information Processing Systems*, volume 15. MIT Press, 2002.
- [13] M. Kirby and L. Sirovich. Application of the karhunen-loeve procedure for the characterization of human faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(1):103–108, 1990.
- [14] H. Lu, Y. Fainman, and R. Hecht-Nielsen. Image manifolds. In *Applications of Artificial Neural Networks in Image Processing III, Proceedings of SPIE*, volume 3307, pages 52–63, 1998.
- [15] K. Pettis, T. Bailey, A. Jain, and R. Dubes. An intrinsic dimensionality estimator from near-neighbor information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1:25–37, 1979.
- [16] R. Pless. Image spaces and video trajectories: Using isomap to explore video sequences. In *Proceedings of the Ninth International Conference on Computer Vision (ICCV)*, pages 1433–1440, 2003.
- [17] M. Polito and P. Perona. Grouping and dimensionality reduction by locally linear embedding. In *Advances in Neural Information Processing Systems*, volume 14. MIT Press, 2002.
- [18] S. Roweis. <http://www.cs.toronto.edu/~roweis/data.html>.
- [19] S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.
- [20] L. Saul and S. Roweis. Think globally, fit locally: Unsupervised learning of low dimensional manifolds. *Journal of Machine Learning Research*, 4(2):119–155, 2003.
- [21] B. Scholkopf, A. Smola, and K.-R. Muller. Nonlinear component analysis and a kernel eigenvalue problem. *Neural Computation*, 10:1299–1319, 1998.
- [22] H. Seung and D. Lee. The manifold ways of perception. *Science*, 290:2268–2269, 2000.
- [23] A. Smola and B. Scholkopf. A tutorial on support vector regression. Technical Report NC2-TR-1998-030, NeuroCOLT2, 1998.
- [24] J. Tenenbaum, V. de Silva, and J. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.
- [25] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.
- [26] P. Verveer and R. Duin. An evaluation of intrinsic dimensionality estimators. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17, 1995.
- [27] K. Weinberger. <http://www.seas.upenn.edu/~kilianw/sde/datasets.htm>.
- [28] K. Weinberger and L. Saul. Unsupervised learning of image manifolds by semidefinite programming. In *Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004.