Transformation-Invariant Embedding for Image Analysis

No Author Given

No Institute Given

Abstract. Image analysis generally and face recognition problem particularly are very challenging in real applications. Appearance-based procedures have been quite influential recently. All of these techniques including Locally Linear Embedding, attempt to recover a low-dimensional characterization of high-dimensional data by measuring Euclidean distances and reconstructions between nearby data points. Unfortunately, Euclidean differences usually do not correspond to natural transformations of typical data, even locally. For example, in image , Euclidean distances rarely correspond to a meaningful perceptual difference between nearby objects. In this research we attempt to improve the quality of manifold learning techniques by modeling local neighborhoods in terms of natural transformations between data—for example, by allowing image operations that extend simple differences and linear combinations. We introduce the idea of modeling local tangent spaces of the manifold in terms of these richer transformations. Given a local tangent space representation, we then embed data in a lower dimensional coordinate system. This leads to improved manifold discovery in natural image data.

1 Introduction

Recently there has been substantial interest in face recognition[1,2,3,4 of 1]. The pattern recognition, computer vision, machine learning and computer graphics communities have been increasingly involved in face recognition, but this problem still remained very challenging in real applications[1 of 1]. Faces are highly variable, the identity of the person, pose, expression and lighting are all sources of variations. In order to extract useful information from such images, a technique requires the ability of underestandind these variabilities.

Both appearance-based (i.e. PCA, ICA, LDA, KPCA, ISOMAP, and LLE), and model-based (i.e. Elastic Bunch Graph matching, Active Appearance Model, and 3D Morphable Model) methods are shown to be suitable for the task of interpreting faces in images (in fact, appearance-based procedures often perform as well as or better than more sophisticated pattern analysis techniques). The appearance-based techniques consider a $n_1 \times n_2$ pixel image as a point in a $n_1 n_2$-dimensional space where each dimension is the intensity at a particular pixel. These technique assume the images to be sampled from an underlying manifold, and then map the data into a single global coordinate system of lower dimensionality where low-dimensional structure of the points coresponds to natural parameterizations of the original image space. The mapping in all of these techniques is reduced to an eigen-value problem.

Particularly Local Linear Embedding (LLE) [**?**] is an approaches that has been quite influential recently. Historically, the main ideas for discovering low dimensional manifolds in high dimensional data has been to find a mapping from the original space to a lower dimensional space that preserves mutual linear reconstruction ability (i.e. principle components analysis [**?**]).

In this case, globally optimal solution is linear manifold. The more recent method for manifold discovery, LLE, is based on exactly the same principle, with the generalization that LLE only seek manifold descriptions that *locally* preserve linear reconstructions. In this way, it avoids recovering linear global solutions [**?**].

There have been many new variants of this idea [**?**,**?**,**?**,**?**]. Although these techniques all produce nonlinear manifolds in different ways, they are generally based on the core assumption that, in natural data, objects can be linearly reconstructed from other data points nearby in Euclidean distance [**?**].

However, this core notion is not universally applicable nor always effective. Particularly, in image data it is easy to appreciate the shortcoming of this idea. For images, weighted linear combinations amount to an awkward transformation whereby source images have their brightness levels adjusted and then are summed directly on top of one another. This is often an unnatural way to capture the image transformations that manifolds are intended to characterize. Figure 1 shows that centered and normalized target images can be fairly well reconstructed from likewise aligned source images, but that even a minor shift, rotation or rescaling will quickly limit the ability of this approach to reconstruct a target image. Similarly, measuring Euclidean distances between images can sometimes be a dubious practice since these distances do not always correspond to meaningful perceptual differences.

Fig. 1. From the left: the first three frames are nearest neighbors, the middle frame is the reconstructed image and the right one is the target. The first row: reconstruction of centered images. Second row: reconstruction of the same images after a shift.

We propose to characterize manifolds locally by identifying local transformations that preserve invariants of the underlying data. That is, we attempt to characterize those transformations that cause points on the manifold to stay on the manifold. More specifically, we approximate the local tangent space around a data object by considering transformations of that object that cause it to stay on (or near) the manifold. Examining transformations that reconstruct neighboring objects from the input data set is one obvious way to achieve this.

2 Local image transformations

For images, it is easy to propose simple local transformations that capture natural invariants in image data better than simply averaging nearby images together. Consider a very simple class of transformations based on receptive fields of pixel neighborhoods: Given an $n1 \times n2$ image x, imagine transforming it into a nearby image $\tilde{x} = T(x, \theta)$, where for each pixel $\tilde{x}_i \in \tilde{x}$ we determine its value from corresponding nearby pixels in x. Specifically, we determine $\tilde{x}_i$ according to

$$\tilde{x}_i = \theta^\top x_{N(i)} \tag{1}$$

where $N(i)$ denotes the set of neighboring pixels of pixel x_i. Thus $T(\cdot, \theta)$ defines a simple local filter passed over the image, parameterized by a single weight vector θ, as shown in Figure 2.

Although this defines a limited class of image transformations, it obviously enhances the image modeling capabilities of weighted image combinations (which are only based on adjusting the brightness level of source images).Many useful types of transformation such as translation, rotation and blurring can be approximated using this simple local transformation. Figure 3 shows that similar images can be much better reconstructed by simple filter transformations rather

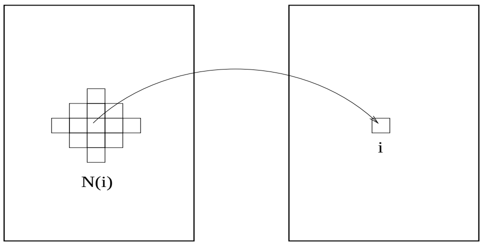

Fig. 2. Illustration of local pixel transformation from left image to right

than merely adjusting brightness levels prior to summing. Here minor translations and appearance changes can be adequately modeled in circumstances where brightness changes fail.

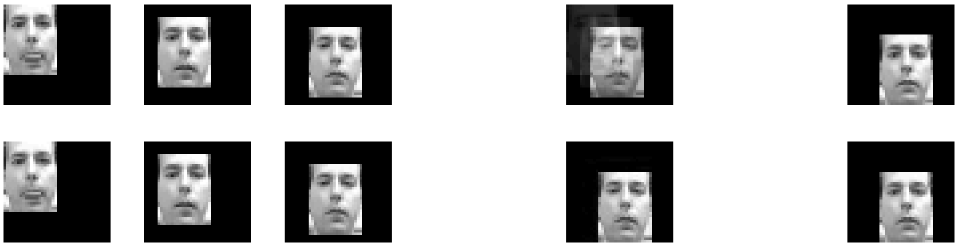

Fig. 3. From the left: the first three frames are nearest neighbors, the middle frame is the reconstructed image and the right one is the target. The first row shows reconstruction by adjusting brightness level. The second row illustrates the reconstruction obtained by filter transformation.

3 Local tangent space modeling

The key to our proposal is to model the local tangent space around high-dimensional data points by a small number of transformations that locally preserve membership in the manifold. Thus, in our approach, a manifold is locally characterized by the invariants it preserves.

We model transformations over the data space by using an operator $T(x, \theta)$ which combines a data object x and a parameter vector θ to produce a transformed object $\tilde{x} = T(x, \theta)$. In general, we will need to assume very little about this operator, but, by making some very simple (and fairly weak) assumptions about the nature of T, we will be able to formulate natural geometric properties that one can preserve in a dimensionality reducing embedding.

First, we assume that T is a *bilinear* operator. That is, T becomes a linear operator on each argument when the other argument is held fixed. Specifically,

$$\begin{aligned} T(ax_1 + bx_2, \theta) &= aT(x_1, \theta) + bT(x_2, \theta) \\ T(x, a\theta_1 + b\theta_2) &= aT(x, \theta_1) + bT(x, \theta_2) \end{aligned} \tag{2}$$

Second, we require the operator to have a local *origin* ω in the second argument that gives an identity map:

$$T(x, \omega) \;=\; x \text{ for all } x \tag{3}$$

With these properties, we can then naturally equate parameterized transformations with tangent vectors as follows. First note that $T(x, \theta) = x + T(x, \delta)$ for $\delta = \theta - \omega$, since by bilinearity we have

$$T(x, \theta) = T(x, \omega + \delta) = T(x, \omega) + T(x, \delta)$$

and also

$$T(x, \omega) = x$$

Thus, we can interpret every transformation of an object x as a vector sum. That is, if $\tilde{x} = T(x, \theta)$ then the difference $\tilde{x} - x$ is just $T(x, \delta)$.

Now imagine transforming a source object x_i to approximate a nearby target object x_j, where both reside on the manifold. The best approximation of x_j by x_i is given by

$$\tilde{x}_{ij} = T(x_i, \tilde{\theta}_{ij})$$

where

$$\tilde{\theta}_{ij} = \arg\min_{\theta} \|x_j - T(x_i, \theta)\|$$

If the approximation error is small, we can claim that the difference vector $\tilde{x}_{ij} - x_i = T(\tilde{\delta}_{ij})$, for $\tilde{\delta}_{ij} = \tilde{\theta}_{ij} - \omega$, is approximately tangent to the manifold at x_i. Consider the norm of the difference vector:

$$\|x_i - \tilde{x}_{ij}\| \;=\; \|T(x, \tilde{\delta}_{ij})\| \;=\; \|\tilde{\delta}_{ij}\| \, \|T(x, \tilde{\eta}_{ij})\|$$

where $\tilde{\eta}_{ij} = \tilde{\delta}_{ij} / \|\tilde{\delta}_{ij}\|$. Here $T(x, \tilde{\eta}_{ij})$ gives the direction of the approximate tangent vector at x_i, and $\|\tilde{\delta}_{ij}\|$ gives the coefficient in direction $\tilde{\eta}_{ij}$. This says that $\tilde{x}_{ij}$ is the projection of x_j onto the tangent plane centered at x_i, since $\tilde{x}_{ij} = x_i + \|\tilde{\delta}_{ij}\| T(x, \tilde{\eta}_{ij})$ is the best approximation of x_j in the local tangent space of x_i.

Intuitively, when we embed x_i and $\tilde{x}_{ij}$ in a lower dimensional space, say by a mapping $x_i \mapsto y_i$ and $\tilde{x}_{ij} \mapsto \tilde{y}_{ij}$, we would like to preserve the coefficient:

$$\|y_i - \tilde{y}_{ij}\| \;\approx\; \|\tilde{\delta}_{ij}\|$$

That is, in the lower-dimensional space, the vector $y_i - \tilde{y}_{ij}$ encodes the embedded direction of the transformation, $T(x_i, \tilde{\eta}_{ij})$, and the length $\|y_i - \tilde{y}_{ij}\|$ encodes the coefficient of the transformation, $\|\tilde{\delta}_{ij}\|$.

4 Transformation-invariant embedding algorithm

Consider t n-dimensional real-valued vectors x_i sampled from some underlying manifold. We can assume each data point and its neighbors lie on, or are close to, a locally linear patch of the manifold. By a linear mapping, consisting of a translation, rotation, and rescaling, the high-dimensional coordinates of each neighborhood can be mapped to global internal coordinates on the manifold. Thus, the nonlinear structure of the data can be identified through two linear steps: first, compute the locally linear patches, and second, compute the linear mapping to the coordinate system on the manifold.

The main goal here is to map the high-dimensional data points to the single global coordinate system of the manifold such that the relationships between points and their neighboring points, generated by proposed local transformation, are preserved. This proceeds in three steps:

In the first step, the neighbors of each data point x_i are identified as follows:

Compute the best point-to-point approximations. The best approximation of x_j by x_i is given by

$$\tilde{x}_{ij} = T(x_i, \tilde{\theta}_{ij})$$

where

$$\tilde{\theta}_{ij} = \arg\min_{\theta} ||x_j - T(x_i, \theta)||$$

The K nearest neighbors of data point x_i are the K best approximations amongst all $\tilde{x}_{ij}$.

The second step of TIE is to reconstruct each data point from its nearest neighbors. Consider a particular data point x_i with K nearest neighbors and reconstruction weights w_j. The reconstruction error can be written as:

$$\varepsilon(W) = \sum_{i=1}^{t} ||x_i - \sum_{j=1}^{K} W_{ij}\tilde{x_{ij}}||^2$$

where the neighbors of x_i are represented by $\tilde{x}_{ij}$ This cost function should be minimized subject to two constraints: first, that each data point x_i is reconstructed only from its neighbors ($W_{ij} = 0$ if $\tilde{x}_{ij}$ is not among the nearest neighbors of x_i). Second, the rows of the weight matrix sum to one ($\sum_j W_{ij} = 1$). This is a constrained least squares problem and, similar to LLE, can be computed in closed form.

The i^{th} data point is reconstructed independently of all others, so the minimization can be done one term at a time. This is equivalent to solving a system of equations with n equations and K unknowns.

Ideally, the solution for the weights should be invariant under the linear mapping from the patch to the global coordinates on the manifold. This ensures that the reconstruction holds equally well in both high-dimensional and low-dimensional space. It can be shown that the weights are invariant to a transformation of the data point and its neighbors if and only if all rows of the weight matrix sum to 1.

Since the optimization can be determined one term at a time, the cost function can be minimized individually for each data point x_i.

$$\varepsilon_i(w) = ||x_i - \sum_{j=1}^{K} w_j \eta_j||^2$$

$$subject\ to \sum_j w_j = 1$$

where η_j denotes neighbors of i^{th} data point. We can rewrite this as:

$$\varepsilon_i(w) = ||x_i - Nw||^2 = ||\chi w - Nw||^2$$

where χ is the matrix of columns x_i repeated K times, and N is the matrix of columns of K nearest neighbors of x_i. Continuing,

$$\begin{aligned} \varepsilon_i(w) &= ||(\chi - N)w||^2 = ((\chi - N)w)^T((\chi - N)w) \\ &= w^T(\chi - N)^T(\chi - N) = w^T G w \end{aligned}$$

where $G = (\chi - N)^T(\chi - N)$. To accommodate the constraint that the weights sum to 1, a Lagrange multiplier, λ, is introduced. Let e be a column vector of ones.

$$\begin{aligned} L(w, \lambda) &= w^T G w + \lambda(w^T e - e) \\ \frac{dL}{dw} &= 0 = 2Gw + \lambda e \\ Gw &= Ce \end{aligned}$$

In practice, we can solve this with C set arbitrarily to 1 and then rescale so w sums to 1.

Like LLE, the third step of TIE is done by choosing the $d-$dimensional vector y_i to minimize the embedding cost function:

$$\Phi(Y) = \sum_{i=1}^{t} ||y_i - \sum_{j=1}^{t} W_{ij} y_j||^2$$

This ensures mutual linear reconstruction ability. Note we are optimizing the coordinates y_i while fixing the weights W_{ij}. This is equivalent to:

$$\Phi(Y) = \sum_{i=1}^{t} ||Y I_i - Y w_i||^2$$

where I_i is the i^{th} column of the identity matrix, and w_i is the i^{th} column of W.

$$\min_{Y} \sum_{i=1}^{t} ||YI_i - Yw_i||^2 =$$

$$\min_{Y} \; trace\; [Y(I-W)(I-W)^T Y^T] = \min_{Y} \; trace(YMY^T)$$

The solution for Y can have an arbitrary origin and orientation. In order to make the problem well-posed, these two degrees of freedom must be removed. Requiring the coordinates to be centered on the origin ($\sum_i y_i = 0$), and constraining the embedding vectors to have unit covariance ($Y^T Y = I$), removes the first and second degrees of freedom respectively.

The cost function can be optimized initially by the second of these two constraints:

$$L(Y, \lambda) = YMY^T + \lambda(YY^T - (N-1)I)$$

$$\frac{dL}{dY} = 0 = 2MY^T + 2\lambda Y$$

$$MY^T = \lambda Y^T$$

L is minimized when the columns of Y^T (rows of Y) are the eigenvectors associated with the lowest eigenvalues of M.

Discarding the eigenvector associated with eigenvalue 0 satisfies the first constraint.

5 Experimental results

We present experimental results on face image data. The first two experiments show the superiority of proposed technique in image analysis in general, and the last experiment illustrates the strength of TIE in face recognition. In all experiments we use the transformation operator on images (1) that was described in Section 2. First, we need to verify that the bilinearity property (2) and origin property (3) are both satisfied. Bilinearity follows trivially from the linearity of (1). Here, the identity transformation $T(x, \omega) = x$ is given by parameters ω that are zero for all neighboring pixels, excepting the center pixel aligned with the target pixel, where the parameter is 1.

Our first experiment is on a single face image. We have conducted experiments on various sets of face images. Although the data is complex, many of these data sets demonstrate manifold behavior over a small number of degrees of freedom. that has been translated vertically.

Figure **??** shows the result of running LLE and Transformation Invariant Embedding (TIE) on the original data shown at the top. This figure shows that the 1-dimensional manifold discovered by LLE is inferior to that discovered by TIE, which had no problem tracking the vertical shift in the image set.

We then conducted an experiment on a database of rotating face images. Figure 5 shows the two-dimensional manifold discovered by LLE, whereas Figure 6 shows the two-dimensional manifold recovered by TIE. In both cases, the first dimension (top) captured the rotation angle of the images, although once again LLE's result is not as good as TIE's. Interestingly, TIE (and to a lesser extent LLE) learned to distinguish frontal from profile views in its second dimension.

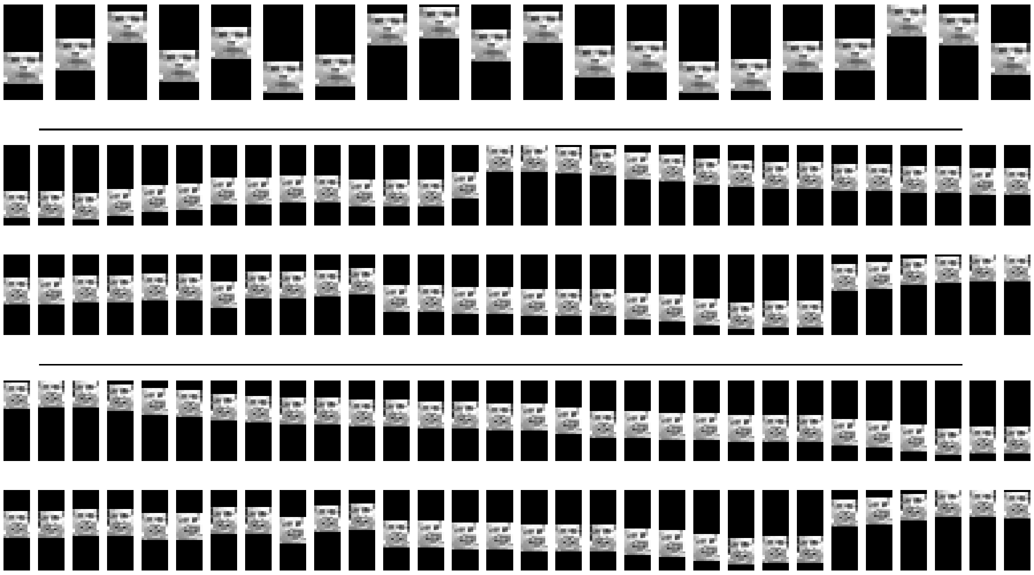

Fig. 4. Top: Original data. 2nd and 3rd rows: 2-dimensional manifold discovered by LLE. 4th and 5th rows: 2-dimensional manifold discovered by TIE

Fig. 5. Two-dimensional manifold discovered by LLE

Finally, we conducted an experiment on a database of face images contains 105 face images of 7 subjects that include variations in both pose, and lighting (see Figure **??**). To reduce computational complexity, each face image was

Fig. 6. Two-dimensional manifold discovered by TIE. Note: first dimension captures rotation, whereas second captures frontal views versus side views

downsampled to 23×29 pixels. The original data space was embedded into three dimensional space. While the first and second dimensions are coresponding to rotation and frontal and profile views, TIE learned to recognize faces in its third dimension. Figure **??** shows the result of running TIE on the original data shown at Figure **??**. Two classes are smeared by TIE and other classes are separated very well. A close look at the third dimension learned by TIE (Figure **??**) shows that even gender can be classified correctly. Figure **??** illustrates that LLE failed in discovering both identity and gender.

6 Conclusion

In many image analysis problems, we know in advance that the data includes different type of transformations. We introduce a way to make standard manifold learning methods such as LLE invariant to transformations in the input. This is achieved by modeling the local tangent space around high-dimensional data points by a small number of transformations that locally preserve membership in the manifold. Thus, in our approach, a manifold is locally characterized by the invariants it preserves.

We model transformations over the data space by using a bilinear operator which produce a transformed object, and show that by making this fairly weak assumptions about the nature of operator, we will be able to formulate natural geometric properties that one can preserve in a dimensionality reducing embedding.

Although our approach is general, we focused on the special case of modeling manifolds in natural image data with emphasis on face recognition problem, and proposed a simple local transformations that capture natural invariants in image data better than simply averaging nearby images together.

We illustrate the algorithms on different examples. We focus on rotation and translation transformation, but other type of transformation such as warping, and out of plane rotation can be used.

References

1. van Leeuwen, J., ed.: Computer Science Today. Recent Trends and Developments. Volume 1000 of Lecture Notes in Computer Science. Springer-Verlag, Berlin Heidelberg New York (1995)
2. Bruce, K., Cardelli, L., Pierce, B.: Comparing object encodings. In Abadi, M., Ito, T., eds.: Theoretical Aspects of Computer Software. Volume 1281 of Lecture Notes in Computer Science., Berlin Heidelberg New York, Springer-Verlag (1997) 415–438
3. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs. 3rd edn. Springer-Verlag, Berlin Heidelberg New York (1996)
4. Baldonado, M., Chang, C.C., Gravano, L., Paepcke, A.: The stanford digital library metadata architecture. Int. J. Digit. Libr. **1** (1997) 108–121