Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Adaptive nonlinear manifolds and their applications to pattern recognition

Hujun Yin *, Weilin Huang

School of Electrical and Electronic Engineering, The University of Manchester, Manchester M60 1QD, UK

ARTICLE INFO

Article history:
Received 21 April 2009
Received in revised form 26 March 2010
Accepted 3 April 2010

Keywords:
Dimensionality reduction
Manifold
Self-organizing map
Nonlinear PCA
Biologically inspired models
Face recognition

ABSTRACT

Dimensionality reduction has long been associated with retinotopic mapping for understanding cortical maps. Multisensory information is processed, fused and mapped to an essentially 2-D cortex in an information preserving manner. Data processing and projection techniques inspired by this biological mechanism are playing an increasingly important role in pattern recognition, computational intelligence, data mining, information retrieval and image recognition. Dimensionality reduction involves reduction of features or volume of data and has become an essential step of information processing in many fields. The topic of manifold learning has recently attracted a great deal of attention, and a number of advanced techniques for extracting nonlinear manifolds and reducing data dimensions have been proposed from statistics, geometry theory and adaptive neural networks. This paper provides an overview of this challenging and emerging topic and discusses various recent methods such as self-organizing map (SOM), kernel PCA, principal manifold, isomap, local linear embedding, and Laplacian eigenmap. Many of them can be considered in a learning manifold framework. The paper further elaborates on the biologically inspired SOM model and its metric preserving variant ViSOM under the framework of adaptive manifold; and their applications in dimensionality reduction with face recognition are investigated. The experiments demonstrate that adaptive ViSOM-based methods produce markedly improved performance over the others due to their metric scaling and preserving properties along the nonlinear manifold.

1. Introduction

Many pattern recognition tasks and hybrid intelligent systems require analysis and exploration of a vast amount of data in order to extract useful information and discover meaningful features, patterns and rules. Clustering data and projecting them onto a lower dimensional space are common practices or components of a hybrid learning system. Dimensionality reduction has long been associated with retinotopic mapping for understanding cortical maps [12]. Multisensory information is processed, fused, fed and mapped to a 2-D cortex in a near-optimal information preserving manner. There exists increasing demand in an increasing number of fields for dimensionality reduction, ranging from high-throughput bioinformatics [20] and neuroinformatics [69], web information science [33], data mining [57], knowledge management [16], information retrieval [31], database indexing [8], to computer vision and image processing [34,52]. The curse of dimensionality has prompted the search for a suitable, smaller and featured representation of a raw data set in order to make data analysis and pattern recognition easier and more efficient. Projecting and abstracting the data onto their underlying or principal subspaces can help reduce the number of features, identify latent variables, detect intrinsic structures, recognize patterns and

* Corresponding author.
E-mail addresses: h.yin@manchester.ac.uk (H. Yin), weilin.huang@postgrad.manchester.ac.uk (W. Huang).

0020-0255/$ - see front matter © 2010 Published by Elsevier Inc.
doi:10.1016/j.ins.2010.04.004

facilitate visualization of variable interactions. With the ever fast increasing data quantity, complexity and dimensionality in many computational tasks, more sophisticated methods are required and are being developed. A great deal of research has been devoted to this emerging topic on improving and extending the capability of classical methods such as principal component analysis (PCA) and multidimensional scaling (MDS) [19,64].

PCA projects a data set onto its principal directions represented by the orthogonal eigenvectors of the covariance matrix of the data. It has long been used for reducing the number of variables and visualizing data in scatter plots or linear subspaces. Singular value decomposition is often adopted to perform the task due to various advantages such as direct operation on the data matrix, stable results even with ill-conditioned data matrix, and decomposition at both feature and data levels. The linearity of PCA, however, limits its power for practical, increasingly large and complex data sets, because it cannot capture nonlinear relationships defined by beyond second order statistics. Extension to nonlinear projection, in principle, can tackle practical problems better; yet a unique solution is yet to be defined [40]. Various nonlinear methods along this line have been proposed, such as the autoassociative networks [32], generalized PCA [25], kernel PCA [51], principal curve and surface [22] and local linear embedding (LLE) [48].

MDS is another popular methodology for extracting data manifold, though no direct mapping function is produced. It projects high-dimensional data points onto a low (often two) dimensional plane by preserving as close as possible the inter-point distances (or pair-wise dissimilarities) [9]. Metric MDS generalizes classical MDS by minimizing a stress function. The mapping is generally nonlinear and can reveal the overall structure of the data. Sammon mapping [49] is a widely known example. In contrast to metric MDS, nonmetric MDS finds a monotonic relationship (instead of metric ones) between the dissimilarities of the data points in the data space and those of their corresponding coordinates in the projected space. More general weighting schemes have been proposed recently and the resulting MDS is called generalized MDS [6]. Isomap [55] applies scaling on geodesic instead of Euclidean distances. MDS methods are generally point-to-point mappings and do not provide a generalizing mapping function or an explicit, generative manifold.

Adaptive neural networks present alternative approaches to nonlinear data projection and dimensionality reduction. They can provide (implicit) generalizing mapping functions. Early examples include feed-forward neural network based mapping [41] and radial-basis-function based MDS [37]. Kohonen's self-organizing map (SOM) [27,30] has become a widely used method for data visualization. Self-organization is a fundamental pattern recognition process, in which intrinsic relationships within the sensory data are learned. SOM is a simplified, abstracted version of Willshaw and von der Malsburg's retinotopic mapping model [59]. Modeling and analyzing such mappings are important to understanding how the brain perceives, encodes, recognizes, and processes the patterns it receives and thus, if somewhat indirectly, are beneficial to machine-based pattern recognition. A recent comprehensive review can be found in [66].

The SOM is topology-preserving vector quantization. The topology-preserving property is utilized to extract and visualize relative mutual relationships among the data. Many variants and extensions have since been proposed such as the generative topographic mapping (GTM) [4] and the visualization induced SOM (ViSOM) [62]. The GTM reformulates the SOM with a mixture of probabilistic model, in which neurons are homoscedastic Gaussians in a latent space and are mapped onto the data space via the expectation–maximization (EM) algorithm. The ViSOM extends the SOM by regularizing the inter-neuron distances within a neighborhood so to preserve (local) distances on the map. The SOM and some variants have been linked with the principal curve and surface (e.g. [47,61]). It has also been widely observed that SOMs produce a similar effect to MDS. The exact connection between SOMs and MDS has recently been established [64,65]. Such connections show the advantages of SOM-based methods in dimensionality reduction. A growing variant of the metric preserving ViSOM has been proposed for embedding nonlinear manifolds.

The remainder of the paper is organized as follows. Section 2 provides a review of existing approaches on nonlinearizing PCA. Section 3 describes MDS and related recent approaches in extracting nonlinear manifolds. Section 4 focuses on SOM, its variants ViSOM and an adaptive growing ViSOM in learning nonlinear, generative manifolds of data sets, as well as their relationship with other approaches. Section 5 applies these manifold methods as dimensionality reduction for face image recognition and compares these performances. Section 6 further discusses various applications of these methods, followed by the conclusions.

2. Nonlinear PCA approaches

PCA is a classical linear projection aiming at finding orthogonal, principal directions in a data set, along which the data exhibits the largest variances. PCA has been widely used for data analysis and dimension reduction and has been the baseline of many advanced nonlinear PCA methods.

2.1. Principal component analysis (PCA)

PCA is obtained by solving an eigenvalue problem of the covariance matrix **C** of a data set, $\lambda\mathbf{V} = \mathbf{CV}$, where columns of **V**, $\mathbf{v}_i$, are eigen vectors. Assume $\mathbf{x} \in R^n$ is of n-dimension and zero mean, then $\mathbf{C} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i\mathbf{x}_i^T$, where N is the total number of data points. By discarding the minor components, PCA can effectively reduce the number of variables and display the dominant ones in a linear subspace of lower dimension. It is the optimal linear projection in the sense of the mean-square error between original points and projected ones, i.e.

$$\min \sum_{\mathbf{x}} \left(\mathbf{x} - \sum_{j=1}^{m} \left(\mathbf{v}_j^T \mathbf{x} \right) \mathbf{v}_j \right)^2 \tag{1}$$

where $\{\mathbf{v}_j, j = 1,2,\ldots,m, m \leqslant n\}$ are the first m principal eigenvectors of the covariance matrix of the data set. The term $\mathbf{v}_j^T\mathbf{x}$ represents the projection of $\mathbf{x}$ onto the jth principal dimension. Efficient and robust statistical methods exist for solving eigenvector problems. Several adaptive learning algorithms have also been proposed for performing PCA such as, the subspace network [46] and the generalized Hebbian algorithm [50]. The limitation of linear PCA is obvious, as it cannot capture nonlinear relationships defined by higher than second order statistics. If the input variables are not linearly correlated, the projection onto a linear principal plane may incur great loss of information.

2.2. Kernel PCA

A nonlinear extension of PCA can be intuitively approached using combined or piecewise local PCA models. That is, the entire input space is partitioned (for example, using a clustering algorithm) into non-overlapping regions and a local PCA can then be formed in each region. However, the extension to nonlinear PCA is not unique due to the lack of a unified mathematical structure and an efficient and robust algorithm, and in some cases due to excessive freedom in selection of representative basis functions [25,40]. Existing methods include the five-layer feed-forward associative network [32] and the kernel PCA [51]. The first three layers of the associative network project the original data onto a curve or surface, providing an activation value for the bottleneck nodes. The last three layers define the curve and surface. The weights of the associative network are determined by minimizing the following objective function,

$$\min \sum_{\mathbf{x}} \left\| \mathbf{x} - \mathbf{f}\{s_f(\mathbf{x})\} \right\|^2 \tag{2}$$

where $\mathbf{f} : R^1 \rightarrow R^n$ (or $R^2 \rightarrow R^n$), the function modeled by the last three layers, defines a curve (or a surface), and $s_f : R^n \rightarrow R^1$ (or $R^n \rightarrow R^2$), the function modeled by the first three layers, defines the projection index.

The kernel-based PCA uses nonlinear mapping and kernel functions to generalize PCA to nonlinear cases. The nonlinear function $\Phi(\mathbf{x})$ maps data onto high-dimensional feature space, where the standard linear PCA can be performed via kernel functions: $k(\mathbf{x},\mathbf{y}) = (\Phi(\mathbf{x}) \cdot \Phi(\mathbf{y}))$. Assuming $\mathbf{x}$ being of zero mean, the projected covariance matrix is then,

$$\mathbf{C}_\Phi = \frac{1}{N} \sum_{i=1}^{N} \Phi(\mathbf{x}_i) \Phi(\mathbf{x}_i)^T \tag{3}$$

The standard linear eigenvalue problem can now be written as [51], $\lambda\mathbf{V} = \mathbf{KV}$, where the columns of $\mathbf{V}$ are the eigenvectors and $\mathbf{K}$ is an $N \times N$ matrix with elements as kernels $K := k(\mathbf{x}_i, \mathbf{x}_j) = \left(\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j) \right)$.

2.3. Local linear embedding

Local linear embedding (LLE) [48] is another way of forming nonlinear principal subspace. The local linearity is defined on a local neighborhood, via the ε ball or k nearest neighbors. Then the linear contributions or weightings, W_{ij}, of these neighboring points are calculated through,

$$\min \sum_{i} \left\| \mathbf{x}_i - \sum_{j=1} W_{ij} \mathbf{x}_j \right\|^2 \tag{4}$$

The embedding is computed via,

$$\min \sum_{i} \left\| \mathbf{y}_i - \sum_{j=1} W_{ij} \mathbf{y}_j \right\|^2 \tag{5}$$

where $\mathbf{y}$ is the embedding coordinates. Both steps can be solved by the least-square method or under the eigen problem.

Laplacian eigenmap [3] forms a local linear mapping by converting the problem to a generalized eigen problem and the solution becomes easily tractable. First, the weightings (of local neighboring points) or heat kernels are constructed,

$$W_{ij} = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{t} \right) \tag{6}$$

Then the embedding is computed via the generalized eigenvalue problem,

$$L\mathbf{f} = \lambda D\mathbf{f} \tag{7}$$

where D is diagonal, $D_{ii} = \sum_j W_{ji}$ and $L = D - W$.

The data is then projected to the subspace spanned by the principal eigen functions ($\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_m$). This approach is also related to spectral clustering [58].

These nonlinear PCA methods, though with different approaches, have many common characteristics. They are all defined on a local neighborhood, thus transforming a global structure to local linear structures. That is, the manifold is constructed on local (linear) graphs. It has been shown that these methods are closely related [21], and can be described under the regularization theory [53].

2.4. Principal curve/surface

The principal curve and surface [22,35,56] are the principled nonlinear extension of PCA. The principal curve is defined as a smooth and self-consistent curve, which does not intersect itself, passing through the middle of the data. Denote $\mathbf{x}$ as a random vector in R^n with density p and finite second moment. Let $f(\cdot)$ be a smooth unit-speed curve in R^n, parameterized by the arc length ρ (from one end of the curve) over $\Lambda \in R$, a closed interval.

For a data point $\mathbf{x}$, its projection index on f is defined as,

$$\rho_f(\mathbf{x}) = \sup_{\rho\in\Lambda}\{\rho : \|\mathbf{x} - f(\rho)\| = \inf_{\vartheta} \|\mathbf{x} - f(\vartheta)\|\} \tag{8}$$

The curve is called the self-consistent principal curve of ρ if,

$$f(\rho) = E[\mathbf{x}|\rho_f(\mathbf{x}) = \rho] \tag{9}$$

The principal component is a special case of the principal curve if the distribution is ellipsoidal. Although it is mainly the principal curve that has been studied, extensions to higher dimensions, e.g. principal surfaces or manifolds are feasible in principle. However, in practice, a good implementation of principal curve/surface relies on an effective and efficient algorithm. The principal curve/surface is more of a concept and practical algorithms are needed for implementation. The HS algorithm is a nonparametric method [22] that directly iterates the two steps of the above definition. It is similar to the standard VQ algorithm combined with some smoothing techniques when only a finite data set is available:

- *Initialization*: Choose the first linear principal component as the initial curve, $f^{(0)}(\mathbf{x})$.
- *Projection*: Project the data points onto the current curve and calculate the projections index, i.e. $\rho^{(t)}(\mathbf{x}) = \rho_{f(t)}(\mathbf{x})$.
- *Expectation*: For each index, take the mean of data points projected onto it as the new curve point, i.e., $f^{(t+1)}(\rho) = E[\mathbf{x}|\rho_{f(t)}(\mathbf{x}) = \rho]$.

The projection and expectation steps are repeated until a convergence criterion is met, for instance, when the change of the curve between iterations is below a threshold.

For a finite data set, the density is often unknown; the above expectation is replaced by a smoothing method such as the locally weighted running-line smoother or smoothing splines. For kernel regression, the smoother is,

$$f(\rho) = \frac{\sum_{i=1}^{N}\mathbf{x}_i\kappa(\rho,\rho_i)}{\sum_{i=1}^{N}\kappa(\rho,\rho_i)} \tag{10}$$

The arc length is simply computed from the line segments. In [2] a modified HS algorithm was proposed by taking the expectation of the residual of the projections in order to reduce the bias. An incremental principal curve was proposed in [26]. It is an incremental, or segment by segment, and arc length constrained method for practical construction of principal curves.

In [56] a semi-parametric model for the principal curve was introduced. A mixture model was used to estimate the noise along the curve; and the EM method was employed to estimate the parameters. Other adaptive learning approaches include the probabilistic principal surfaces [7].

3. Multidimensional scaling approaches

Multidimensional scaling (MDS) is a traditional subject related to dimension reduction and data visualization. MDS aims to embed high dimensional data points onto a low dimensional (often 2-or 3-D) plane by preserving as closely as possible inter-point metrics [9]. The projection, which is often calculated via an optimization process of a stress function, is generally nonlinear and can reveal the overall structure of the data. Recent extensions see the use of neighborhood to confine the stress locally and the use of adaptive learning algorithms instead of an optimization method.

3.1. Classical, metric and nonmetric MDS

Let δ_{ij} denote the dissimilarity between data points $\mathbf{x}_i$ and $\mathbf{x}_j$. δ_{ij} is often calculated (but not necessarily) by the Euclidean distance of data vectors $\delta_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|$. Let $\mathbf{y}_i$ and $\mathbf{y}_j$ be the mapped points or coordinates of points $\mathbf{x}_i$ and $\mathbf{x}_j$ in the visual space, then the distance between the mapped points is $d_{ij} = \|\mathbf{y}_i - \mathbf{y}_j\|$.

Classical MDS looks for a configuration so that the distances between projected points match the original dissimilarities, i.e. $d_{ij} = \delta_{ij}$, $\forall i,j$. *Metric MDS* seeks that dissimilarities are proportional to the distances of projected points, $d_{ij} = f(\delta_{ij})$, $\forall i,j$;

where f is a continuous, monotonic function, or a metric transformation function that transforms the dissimilarities to distance metrics. In practice, exact dissimilarity-distance matches may not be possible due to data noise and imprecision, the equality is replaced by approximation, i.e. "≈", meaning "as equal as possible" [5].

An MDS configuration is often sought by minimizing the following general cost, or the raw *Stress*, function,

$$S = \sum_{i,j} (f(\delta_{ij}) - d_{ij})^2 \tag{11}$$

In some cases, the above raw stress is normalized by $\sum_{i,j} d_{ij}^2$ or $\sum_{i,j} \delta_{ij}^2$ to give a relative reading of the overall stress. Other normalization schemes are also available. For example, in Sammon mapping [49] an intermediate normalization is used to preserve good local distributions and at the same time to maintain a global structure,

$$S = \frac{1}{\sum_{i<j} \delta_{ij}} \sum_{i<j} \frac{(\delta_{ij} - d_{ij})^2}{\delta_{ij}} \tag{12}$$

The Newton optimization method is used to recursively solve for the optimal configuration.

For general metric MDS, especially when original dissimilarities need to be transformed to a distance like form, f is a monotonic transformation function, e.g. a linear function. For classical MDS and many cases of metric MDS, f is simply the identity function, and the stress becomes,

$$S = \sum_{i,j} (\delta_{ij} - d_{ij})^2 \tag{13}$$

That is, metric MDS configuration tries to preserve, as faithfully as possible, the pair-wise distances of the original data points on the projected space. PCA is a special configuration of such MDS.

Nonmetric MDS deals with the rank order of the dissimilarities and seeks a configuration such that distances between pairs of mapped points match order-wise "as well as possible" the original dissimilarities. That is, a nonmetric MDS looks for a projection function f that is a monotonic function and satisfies,

$$\text{if} \quad \delta_{ij} \leqslant \delta_{kl}, \quad \text{then} \quad d_{ij} \leqslant d_{kl}, \quad \forall i,j,k,l \tag{14}$$

Nonmetric MDS produces an ordinal scaling rather than a metric one.

A drawback of MDS is the lack of an explicit projection function, as MDS is usually a point-to-point mapping and cannot *naturally* accommodate new data points. Thus for any new input data, the mapping has to be recalculated based on all available data. Although some methods have been proposed to accommodate new arrivals using triangulation [10,36], the methods are generally not adaptive.

3.2. Adaptive MDS

The drawbacks of traditional MDS can be overcome by implementing or parameterizing MDS using neural networks. In [41], a feed-forward network was used to parameterize the Sammon mapping and an unsupervised training method was derived to train the network. The derivation is similar to the back-propagation algorithm, by minimizing the Sammon stress instead of the total errors between desired and actual output. In [37] a radial-basis function (RBF) network was used to minimize a simple stress function, in order to perform MDS.

In [55], the Isomap was proposed to use geodesic (curvature) distance (instead of Euclidean) for better scaling of nonlinear manifolds. The geodesic distance along the manifold was calculated (or cumulated) via neighborhood graphs or neighboring points. Selecting a suitable neighborhood size can be a difficult task and often needs a cross-validation procedure. The Isomap has been reported as being unstable [1].

It is worth noting that while MDS minimizes the difference between the dissimilarities in the original and mapped spaces, the c measure proposed as a unified objective for topographic mapping [18] maximizes the correlation between the two. One can argue that when the dissimilarities are normalized, the two are equivalent. Curvilinear component analysis (CCA) [11] is another extension of MDS. It detects the intrinsic geometric properties of data by preserving local distance relationships via minimizing an error function defined as,

$$E = \frac{1}{2} \sum_{i} \sum_{j \neq i} (\delta_{ij} - d_{ij})^2 \varphi(d_{ij}, \theta) \tag{15}$$

where $\varphi(d_{ij},\theta)$ is a monotonically decreasing neighborhood function with respect to the distance in the projected space and is used for preserving local topology and maintaining shorter distances than longer ones; and θ is its parameter.

4. SOM approaches

Kohonen's self-organizing map (SOM) is an abstract, simplified mathematical model of the mapping between nerve sensors and the cerebral cortex [27,30]. Modeling and analyzing such mappings are helpful to understanding how the brain perceives, encodes, recognizes, and processes the information it receives and thus are also beneficial to machine-based pattern recognition. External stimuli are received by various sensory or receptive fields, coded, combined and abstracted by the living neural networks, propagated through axons, and projected onto the cerebral cortex, often to distinct parts of the cortex. Different areas of the cortex (cortical maps) respond to different sensory inputs, though many functions and actions require collective responses from various areas. Topographically ordered mappings are widely observed in the cortex. The main structures (primary sensory areas) of the cortical maps are established genetically in a predetermined manner [28]. More detailed areas (associative areas) between the primary sensory areas, however, are developed through self-organization gradually during life and in a topographically meaningful fashion. Therefore, studying such topographic projections, which had been ignored during the early period of neural network research, is undoubtedly fundamental for effective representation of sensory information and feature extraction.

4.1. SOM as nonmetric-scaling manifold

Von der Malsburg and Willshaw earlier developed in mathematical form, the self-organizing topographic mappings, mainly from two-dimensional presynaptic sheets to two-dimensional postsynaptic sheets, based on retinatopic mapping: the ordered projection of visual retina to visual cortex [59]. Kohonen abstracted this self-organizing learning model and proposed a much simplified mechanism which ingeniously incorporates the Hebbian learning rule and lateral interactions [27]. This simplified model can emulate the self-organization effect.

In the SOM, a set of neurons, often arranged in a 2-D rectangular or hexagonal grid or lattice, is used to form a discrete, topological mapping of an input space, $\mathbf{X} \in R^n$. All the weights $\{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_M\}$ are initialized to either small random numbers or specific values (e.g. principal subspace), where $\mathbf{w}_i$ is the weight associated to neuron i and is a vector of the same dimension, n, of the input, and M is the total number of neurons. Denote $\mathbf{r}_i$ as the discrete vector defining the position (coordinates) of neuron i on the map grid. Then the algorithm iterates the following steps.

- At each time t, present an input, $\mathbf{x}(t)$, select the winner,

$$v(t) = \arg\min_{k \in \Omega} \|\mathbf{x}(t) - \mathbf{w}_k(t)\| \tag{16}$$

- Update the weights of the winner and its neighbors,

$$\Delta \mathbf{w}_k(t) = \alpha(t)\eta(v,k,t)[\mathbf{x}(t) - \mathbf{w}_k(t)] \tag{17}$$

- Repeat until the map converges,

where $\alpha(t)$ is the learning rate, $\eta(v,k,t)$ the neighborhood function, and Ω the set of neuron indexes. Although one can use a top-hat type of neighborhood function, a Gaussian, $\eta(v,k,t) = \exp\left(-\frac{d_{vk}^2}{2\sigma(t)^2}\right)$, is often used in practice with $\sigma(t)$ representing the effective range of the neighborhood at time t and $d_{vk} = \|\mathbf{r}_v - \mathbf{r}_k\|$, the distance between neurons v and k on the map grid.

The SOM was proposed to model the sensory-to-cortex mapping, and is thus an associative memory mechanism. The SOM has been shown to be an asymptotically optimal VQ [67]. More importantly, with the neighborhood learning, the SOM is an error tolerant VQ and Bayesian VQ [38,39]. SOM has been linked with minimal wiring of cortex-like maps [12,42].

For data visualization, however, the inter-neuron distances, when referred to the data space, have to be crudely or qualitatively marked by colors or grey levels on the trained map. The coordinates of the neurons (the result of scaling) are fixed on the lower dimensional (often 2-D) grid and do not resemble the distances (dissimilarities) in the data space.

4.2. ViSOM as metric-scaling manifold

For metric scaling and visualization, a direct and faithful display of data structure and distribution is highly desirable. The ViSOM extends the SOM for distance preservation on the map [62]. In order to achieve that, the updating force, $[\mathbf{x}(t) - \mathbf{w}_k(t)]$, of the SOM algorithm is decomposed into two elements $[\mathbf{x}(t) - \mathbf{w}_v(t)] + [\mathbf{w}_v(t) - \mathbf{w}_k(t)]$. The first term, $[\mathbf{x}(t) - \mathbf{w}_v(t)]$, represents the updating force from the winner v to the input $\mathbf{x}(t)$, and is the same as the updating force used by the winner v. The second term, $[\mathbf{w}_v(t) - \mathbf{w}_k(t)]$, is a lateral contraction force bringing neighboring neuron k to the winner. In the ViSOM, this lateral contraction force is regulated in order to help maintain unified inter-neuron distances locally on the map. The update rule with the simplest constraint is,

$$\Delta \mathbf{w}_k(t) = \alpha(t)\eta(v,k,t)([\mathbf{x}(t) - \mathbf{w}_v(t)] + \beta[\mathbf{w}_v(t) - \mathbf{w}_k(t)]) \tag{18}$$

where β is a constraint coefficient-the simplest form being $\beta = \delta_{vk}/(d_{vk}\lambda) - 1$, δ_{vk} is the distance of neurons' weights in the input space, d_{vk} is the distance of neurons' indexes on the map, and λ is a resolution constant. A further refresh step (using neurons' weights as the input) is added to ensure a smooth expansion of the map in areas where the data is sparse or empty [61,62].

The ViSOM regularizes the inter-neuron contraction so that local distances between the nodes on the map are analogous to the distances of their weights in the data space. In addition to SOM's objective to minimize the quantization error, the aim is also to maintain constant inter-neuron distances locally. When the data points are eventually projected onto the trained map, the distance between data points i and j on the map is proportional to the distance of these two points in the data space, at least locally, subject to the quantization error (the distance between a data point and its neural representative). That is, $d_{ij} \propto \delta_{ij}$or $\lambda d_{ij} \approx \delta_{ij}$. This makes data visualization more direct and quantitatively measurable. The resolution of the map can be enhanced by interpolating a trained (small) map or by incorporating the local linear projection (LLP) method [63]. Instead of projecting onto the winning node v (or $\mathbf{w}_v$), the data point $\mathbf{x}$ is projected to the sub plane spanned by the two closest edges. The projected point is therefore,

$$\mathbf{x}' = \mathbf{w}_v + \max_{v'=v\pm1}\left\{\frac{(\mathbf{x}-\mathbf{w}_v)\bullet(\mathbf{w}_v-\mathbf{w}_{v'})}{\|\mathbf{w}_v-\mathbf{w}_{v'}\|^2}, 0\right\} \tag{19}$$

where '•' denotes dot-product.

The size or covering range of the neighborhood function decreases from an initially large value to a final small one. The final neighborhood can be made adaptive. The rigidity or curvature of the map is controlled by the size of the neighborhood. The larger this size the flatter the final map is in the data space.

Several improvements have since been made to the ViSOM for improved stability and flexibility [13,60]. A growing ViSOM (gViSOM) has been proposed recently to effectively extract a highly nonlinear manifold [65].

The similarities between SOMs and MDS in terms of topographic mapping, mostly the qualitative likeness of the mapping results, have been reported before [65]. However limitations of using the SOM for MDS have also been noted [15] – the main one being that SOM does not preserve distance. Many applications combine the SOM and MDS for improved visualization of the SOM projection results. In [61], it is shown that the distance-preserving ViSOM approximates a discrete principal manifold. One advantage of the ViSOM is that its neighborhood is usually made adaptive according to data characteristics in various regions, unlike that in other nonlinear PCA or manifold methods a preset neighborhood size has to be set empirically. ViSOM has also been shown to produce a similar mapping result as to metric MDS [64,65]. ViSOM is a metric MDS, while order-preserving SOM is a kind of nonmetric MDS.

It has also been shown that SOM-based algorithms are difficult to converge to highly nonlinear manifolds when using a map of pre-fixed size [65]. To improve the local distance-preserving capability of ViSOM, an incremental or growing ViSOM (gViSOM) has been proposed [65] for embedding and metric-scaling nonlinear manifolds. Details of the adaptive gViSOM algorithm are as follows:

1. Start with a small initial map (e.g. 5 × 5), either rectangular or hexagonal. Place the initial map onto a linear subspace of either the entire or a local region of the data space. Set the desired resolution and the neighborhood size (locality).
2. Randomly draw a sample from the data space and find the winning neuron with the shortest distance.
3. If the sample falls within the neighborhood, update the weights of the neurons of the neighborhood using the ViSOM algorithm; otherwise go back to step 2.
4. At regular iteration intervals (e.g. 2000 iterations), if the growing condition is met (that is, the data is underrepresented by the existing map), grow the map by adding a column or row to the side with the highest activities (measured by the winning frequencies). The added column or row is a linear extrapolation of the existing map. Other growing structures can be used, such as incrementing polygons instead of entire columns or rows for a free structure of the map and efficient use of neurons.
5. As in the ViSOM, at regular intervals (every certain number of iterations), refresh the map (neurons) probabilistically.
6. Check if the map has converged. If not go back to step 2; if so go to the next step.
7. Project the data samples onto the map, either to the neurons or by the LLP resolution enhancement.

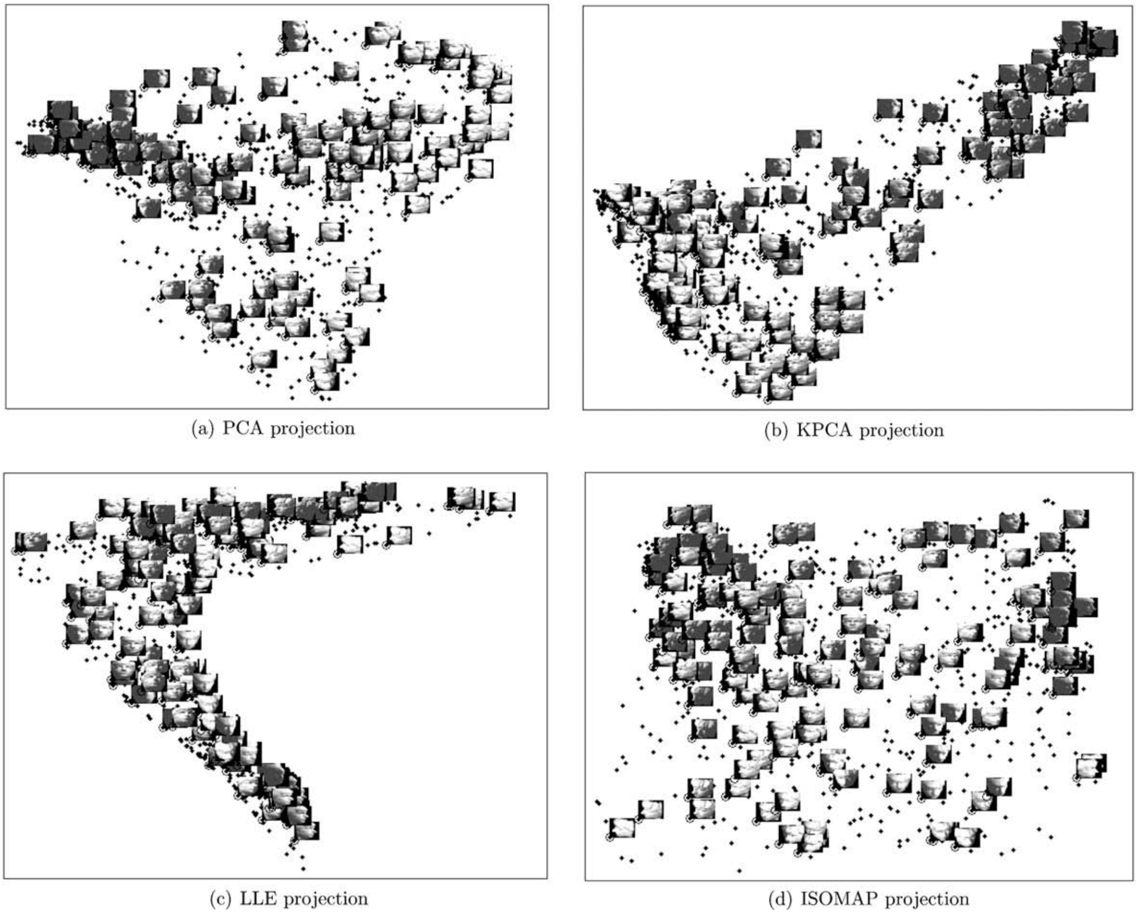
4.3. Other SOM related approaches

Other SOM-based or motivated manifold approaches include adaptive subspace SOM (ASSOM) [29], the GTM [4], self-organizing mixture network (SOMN) [68] and topological product of experts [17]. These methods model the data by a means of a latent space. They belong to the semi-parameterized mixture model, although types and orientations of the local distributions vary from method to method. These approaches also resemble or can produce nonlinear PCA. SOM is used to quantize and segment the input space into local regions (the so-called Voronoi tessellation) and local probabilistic models formed in these regions can interpret PCA locally. Other similar neural approaches include the early elastic net [12].

5. Adaptive nonlinear manifolds for face recognition

5.1. Two-dimensional manifold representation of face data

The manifold methods mentioned above have been investigated for dimensionality reduction and feature extraction on face image data. The first experiment is on mapping a set of pose varying faces of a single subject onto a two-dimensional manifold. The data set is obtained from [55]. The results of various methods, PCA, Kernel PCA, LLE, Isomap, GTM, SOM and gViSOM, are shown in Figs. 1 and 2.



(a) PCA projection

(b) KPCA projection

(c) LLE projection

(d) ISOMAP projection

Fig. 1. Two-dimensional representations of facial images with pose and lighting variations of a single subject by PCA (a), KPCA (b), LLE (c) and ISOMAP (d). The Gaussian radial-basis function with radius of 4 is used for kernel PCA. The size of neighborhoods used by LLE and ISOMAP is 8.

The figures show that two-dimensional projections of LLE, ISOMAP and gViSOM capture better intrinsic structure of pose variations than other projections. Smooth transitions of pose change and lighting can be observed in LLE, ISOMAP and gViSOM projections. However, PCA, KPCA, SOM and GTM projections are more sensitive to lighting impact than pose variations, leading to some degree of overlap for those poses.

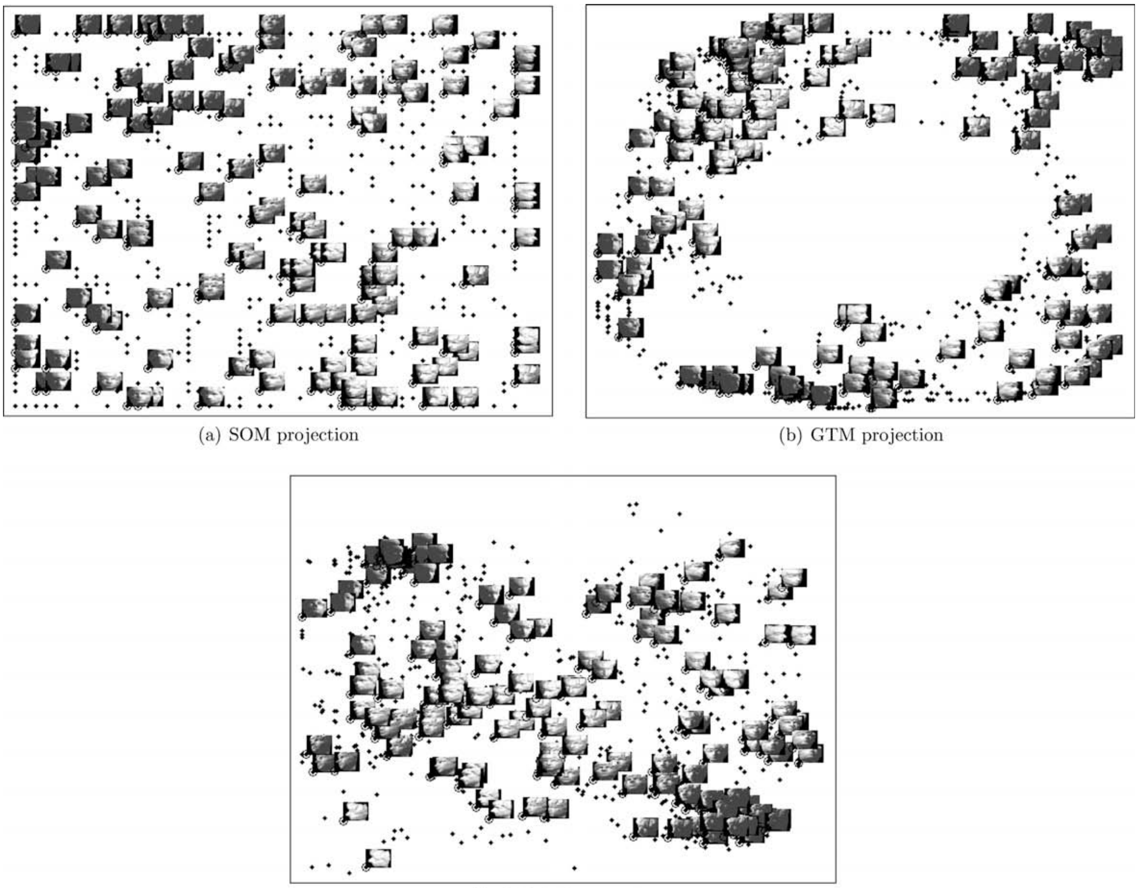
5.2. Dimensionality reduction for face recognition

For face recognition or classification, three common classifiers were used in the experiment: the Nearest-Neighbor (NN), soft *k*-Nearest Neighbor (soft *k*-NN) [54] and the Linear Discriminant Analysis (LDA) [14]. NN is the simplest classifier which assigns a test sample to the class of the most similar example in the training set. In soft *k*-NN classifier, each principal component outputs a confidence value, which gives the degree of support for the component in every face representation, and then the final decision is given by considering all of these confidence values.

LDA is an efficient and widely used linear classifier. It tries to find the linear projection of the data set that minimizes the within-class scatter while maximizing the between-class separation. The ratio of the determinant of the between-class scatter matrix and the within-class scatter matrix in the projected space is maximized by solving an eigenvalue problem.

In the experiment, various manifold methods were used for dimension reduction in the preprocessing of raw face images, and then each of the classifiers was used for classification. The performances of the dimension reduction methods were evaluated and compared based on the same classifier. The experiment was conducted on a publicly available database, the ORL database (of Olivetti Research Laboratory), which consists of 40 subjects with 10 different face images for each subject. All images in the database were taken against a dark homogeneous background with an up-right, frontal position and have the same size of 92×112. Face images vary in terms of lighting conditions, facial expressions or facial details.

In PCA, nonlinear PCA and MDS-based methods, the number of dimensions (92×112) of entire face images was reduced to 60 (larger numbers or further reductions do not improve the performance significantly in this case [24]). Two types of kernel-PCA, *polynomial* (KPCA1) and *Gaussian Radial Basis* (KPCA2), were used with degree of 2 and radius of 30, respectively.



(a) SOM projection

(b) GTM projection

(c) gViSOM projection

Fig. 2. Two-dimensional representations of facial images with pose and lighting variations of a single subject by SOM (a), GTM (b) and gViSOM (c). The size of SOM and GTM is 50 × 50, and the resolution of gViSOM is 1.1.

The size of neighborhood used by LLE was set to 30. The 60-dimensional face representations (vectors) were then used for training and testing the NN, soft *k*-NN and LDA classifiers. These parameters were chosen on the best results obtained.

In block-based methods, images were divided into small blocks (5 × 5), then the reduction was achieved by quantizing these blocks. This is the so-called vector quantization (VQ), a standard approach commonly used in SOM-based dimension reduction and image compression. In SOM-based experiments for dimension reduction, the face images were first locally sampled by moving a window of size 5 × 5 over the entire image by four pixels each time. The sampled images were reconstructed to the size of 25 × 23 × 28 after sampling. That is, each sampled face image contains 23 × 28 25-dimensional subsamples. These 25-dimensional samples were used as the input for training SOM-based manifolds. For each method, the size and parameters have been optimized for the best performance. For example, the sizes of SOM, ViSOM and gViSOM varied from 5 × 5 to 30 × 30, and the chosen sizes represent the cases with the best performances (i.e. 30 × 30 for all methods of pre-fixed size such as SOM, GTM and ViSOM. δ was set to 0.5 for ViSOM. For gViSOM, its size grew from 5 × 5 and ended in 16 × 19 as the performance is already much better than the others). Then all 25-dimensional samples of each face image were passed through the trained SOM, ViSOM and gViSOM, and represented by the 2-D index values of the corresponding winners on the maps (as shown in Figs. 3–5g and h), representing the projections of the faces on the maps. Each dimension of the face projection can be reconstructed as a feature face (of size of 23 × 28, examples of two subjects are shown in Figs. 3–5b, c, e and f), which resemble features of the original face images. As can be seen, ViSOM methods produce better feature faces (i.e. resemble more the original faces) due to its metric preserving property in feature extraction.

For an objective evaluation, the performances of SOM-based and PCA-based methods were investigated on the same classifier for each experiment on all subjects of the ORL database. The number of training images was varied from 3, 4, 5 to 6 per subject and the remaining 7, 6, 5, and 4 were used as test images, respectively. The results reported are the average results of 10 independent implementations with different randomly chosen training images. Meanwhile, the same choices of training (and test) images were used by all the methods to ensure an unbiased comparison. The results of PCA-based methods followed by the NN, soft *k*-NN or LDA classifier are shown in Table 1. The performances of SOM-based methods with the NN or soft *k*-NN classifier are listed in Table 2.

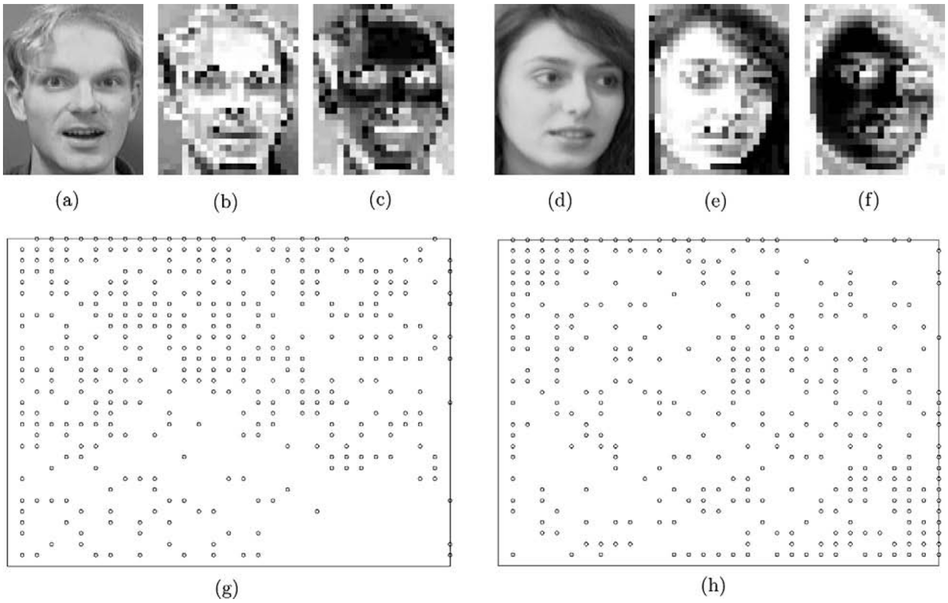


Fig. 3. Examples of SOM projection, (a) and (d) original faces; (b) and (e) feature faces from *x*-axis of the map; (c) and (f) feature faces of *y*-axis of the map; (g) and (h) projections of face blocks on the map. ((b), (c), (e) and (f) are rescaled for display.)

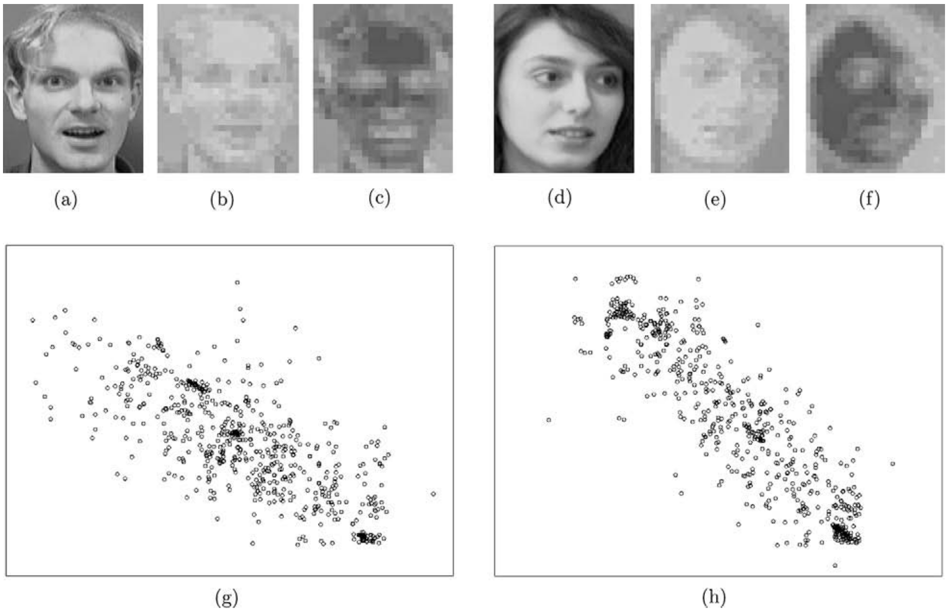


Fig. 4. Examples of ViSOM projection, (a) and (d) original faces; (b) and (e) feature faces from *x*-axis of the map; (c) and (f) feature faces of *y*-axis of the map; (g) and (h) projections of face blocks on the map. ((b), (c), (e) and (f) are rescaled for display.)

The tables show that with more training samples, error rates decrease in all methods as expected. Both PCA-based and SOM-based methods used the same ORL database. All the reduced vectors (in PCA-based methods) or blocks (in SOM-based) were then used to train the classifiers. Tables 1 and 2 are separate results for two groups of methods (PCA-based and SOM-based). In the first set, it shows that PCA is almost the best with the NN classifier, while LLE is the best with*k*-NN and LDA. In the second group, ViSOM, esp. gViSOM, is the best with either NN or *k*-NN classifiers. Two groups of results may not be directly comparable, unless both sets of methods are tuned to have the same reduction ratio. But, at least they serve as a reference. The SOM-based methods have similar or better performances than PCA-based methods; the ViSOM and gViSOM yield markedly improved results. With the soft *k*-NN classifier, LLE has slightly lower error rates than other PCA-based methods with error rates reaching 3.75% and 2.69% by LDA classifier on 5 and 6 training images, respectively. The error rates of gViSOM with LLP in training five and six faces are as low as 2.1% and 0.75%, respectively.

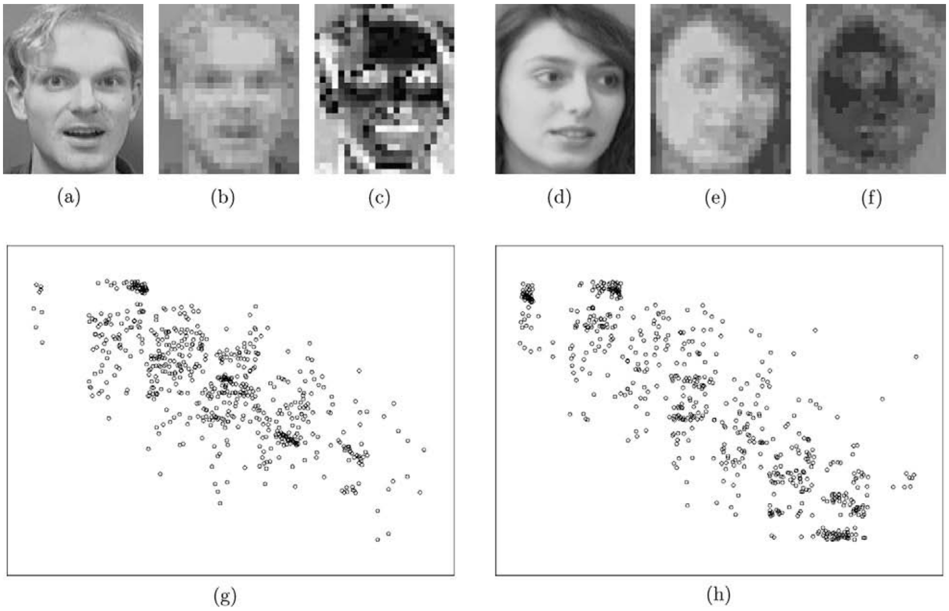


Fig. 5. Examples of gViSOM projection, (a) and (d) original faces; (b) and (e) feature faces from *x*-axis of the map; (c) and (f) feature faces of *y*-axis of the map; (g) and (h) projections of face blocks on the map. ((b), (c), (e) and (f) are rescaled for display.)

Table 1
Error rates of PCA-based methods followed by a NN, soft *k*-NN or LDA classifier.

No. of training faces	Error rates (%)					
	PCA	KPCA1	KPCA2	LLE	ISOMAP	CCA
	NN classifier					
3	13.25	13.54	12.25	**11.25**	14.21	12.25
4	8.08	8.64	**7.17**	7.29	8.54	8.17
5	**5.65**	5.75	5.80	5.70	6.85	5.80
6	**3.56**	**3.56**	4.06	4.13	4.13	4.06
	Soft k-NN classifier					
3	13.43	15.50	11.75	**11.29**	14.14	12.68
4	8.69	9.42	9.08	**7.25**	8.79	8.46
5	6.15	6.45	7.00	**5.50**	6.90	5.85
6	4.26	4.50	5.87	3.94	4.75	**3.81**
	LDA classifier					
3	**9.36**	11.07	10.07	9.71	13.46	12.15
4	**5.08**	6.00	5.96	6.75	8.37	7.21
5	3.80	4.15	4.95	**3.75**	6.90	5.40
6	3.12	3.31	3.19	**2.69**	4.31	4.31

Table 2
Error rates of SOM-based methods followed by a NN or soft *k*-NN classifier.

No. of training faces	Error rates (%)					
	GTM	SOM	ViSOM	ViSOM*	gViSOM	gViSOM*
	NN classifier					
3	17.82	13.71	10.61	**10.50**	10.86	10.79
4	13.63	8.79	6.42	**6.37**	6.46	6.54
5	10.75	6.95	**4.30**	4.35	4.40	4.50
6	8.31	4.56	2.69	**2.63**	2.75	2.88
	Soft k-NN classifier					
3	15.46	7.71	7.75	7.32	7.21	**6.71**
4	11.58	4.21	3.88	3.79	**3.67**	**3.67**
5	9.40	2.75	2.80	2.40	2.55	**2.10**
6	7.13	1.75	1.25	1.19	0.81	**0.75**

6. Other applications and discussion

Dimension reduction methods and manifolds have found numerous applications in a number of fields. A large number of articles can be found in the literature. Here, we list a sample of further typical applications.

6.1. Data visualization

Data visualization looks for low dimensional, visual subspaces of the data space in order to observe and display characteristics of the data along these subspaces. Adaptively extracting data's manifold or submanifolds is a principled way to visualize the data set in low dimensional spaces. The approaches described in the previous sections serve as good tools for such applications. For example, SOM and its variants have been widely used for visualizing and organizing sociological data of high-dimensional attributes, for instance web documents using WEBSOM) [33] and unstructured text documents using a file-explorer alike tree-structure 1-D SOMs [16]. In visualizing and organizing non-numerical type of data such as text and letter sequences, the ordinal relationship among the data items are important. Therefore, topology-preserving mapping such as SOMs are useful and effective.

For more numerical visualization, a metric preserving mapping such as metric MDS or nonlinear PCA is essential. ViSOM has been used for visualizing this kind of data, in particular their distribution and structure [61] as it can extract highly nonlinear metric manifolds [65].

6.2. Embedding and generative models

Fitting high-dimensional data to a low dimensional model can produce an effective generative model of the data. Such models can be used for finding principal, nonlinear latent variables thus underlying models of the data set, as well as filtering noise in the data. Many methods in Sections 2 and 4 are useful tools in this aspect. For instance, Isomap, LLE, GTM and gViSOM have been used to extract highly nonlinear generative, metric manifolds.

6.3. Pattern recognition

In many pattern recognition tasks, the first step is to reduce the number of features or dimensionality of the data. For example, both PCA and SOM have been tested previously on reducing image dimensionality in a face recognition task and nonlinear SOM has been shown to outperform linear PCA [34]. Isomap and LLE have been applied to extract the most important (nonlinear) variations such as pose and lighting in face database decomposition [48,55].

6.4. Feature selection and reduction

Usually, feature selection or reduction is performed by supervised learning. However, feature reduction can be carried out in an unsupervised way using a dimensionality reduction method. More recently, semi-supervised methods, which combine supervised and unsupervised procedures, have become an active area of research and have been used for more effective use of both labeled and unlabeled data. For example, a Laplacian score has been proposed to downsize and select features [23].

6.5. Bioinformatics and neuroinformatics

In bioinformatics, there is a huge demand for analyzing the effect of genes under certain conditions or identifying a subset of a genome that contribute to certain biological functions or malfunctions. The problem is particularly challenging when the number of genes under study is much greater than the number of samples. Microarray data analysis has become increasingly reliant on dimensionality reduction techniques and manifold clustering. For instance, in [44], a hybrid dimension reduction method is employed prior to the classification of tumor tissue samples. In [43], a 1-D SOM is utilized to efficiently group yeast *Saccharomayces cerevisiae* cell cycle data, in combination with a novel temporal shape metric.

Neuroinformatics and systems neuroscience also rely intensively on computational techniques for revealing the interaction between neurons as well and between neuronal populations or networks. In light of large numbers of trials and sampling points, decoding experimental data sets becomes particularly challenging. In [69], a topological mapping is applied to the spike trains recorded in rat somatosensory cortex in response to vibrissal stimulations. Combined with the information theoretic framework, it has effectively shown the neuronal response to be an energy code.

6.6. Time series and sequence analysis

Temporal or spatio-temporal manifold is a challenging new direction in this area. Most existing dimension reduction methods regard data as spatial vectors. Such methods may be adopted for temporal signals or sequences when considering time series as vectors of consecutive time points (e.g. using a sliding window over the time series). However, such an approach is not optimal as spatial vector representation does not take into account the temporal relations of time points and the commonly used Euclidean metric does not consider any temporal information. New approaches that naturally consider temporal input are needed. For example, when a temporal metric can be used, then the spatial method can turn into a temporal method [43]. More recently, a self-organizing mixture autoregressive (SOMAR) network has been proposed to cluster and model nonstationary time series [45]. The method regards the entire nonstationary time series as a hybrid of local linear regressive models and it has been shown to produce better forecasting results for financial data than many spatial methods and the GARCH model.

7. Conclusions

In this paper, an overview of nonlinear principal manifolds for dimensionality reduction and a study on their applications for pattern recognition, esp. face recognition, are presented. The existing methods have been categorized into three groups: nonlinear PCA, MDS-based, and SOM-based. While nonlinear PCA approaches are particularly advantageous, especially when they are framed under the kernel method, as the problem is then converted to a linear eigenvalue problem and a unique solution can be expected, they still need to set a number of kernel or geometrical parameters. Usually, they are not adaptive learning methods. Adaptive learning approaches such as SOM-based can gradually extract low dimensional manifolds. SOM-based methods have been shown to be either nonmetric or metric MDS. The flexibility and diversity of these adaptive learning approaches make them widely applicable and integrateable with other computational paradigms. Their application for dimensionality reduction in face recognition has been studied in detail. In certain cases, performance differences between linear and nonlinear methods are marginal. The experimental results show that the adaptive manifolds based on ViSOM or gViSOM outperform consistently various other SOM-based methods. It is largely due to the fact that ViSOM is an adaptive and metric preserving manifold. Various other applications of dimensionality reduction and manifolds have also been reviewed, together with some challenges ahead. Research and application in nonlinear manifolds are set to flourish in many disciplines in this data-rich era.

Acknowledgements

The authors thank the reviewers and editors for their valuable comments and suggestions.

References

[1] M. Balasubramanian, E.L. Schwartz, The Isomap algorithm and topological stability, Science 295 (2002) 7a.
[2] J.D. Banfield, A.E. Raftery, Ice floe identification in satellite images using mathematical morphology and clustering about principal curves, J. Am. Stat. Assoc. 87 (1992) 7–16.
[3] M. Belkin, P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, Neural Comput. 15 (2003) 1373–1396.
[4] C.M. Bishop, M. Svensén, C.K.I. Williams, GTM: the generative topographic mapping, Neural Comput. 10 (1998) 215–235.
[5] I. Borg, P.J.F. Groenen, Modern Multidimensional Scaling: Theory and Applications, second ed., Springer, 2005.
[6] A.M. Bronstein, M.M. Bronstein, R. Kimmel, Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching, PNAS 103 (2006) 1168–1172.
[7] K.-Y. Chang, J. Ghosh, A unified model for probabilistic principal surfaces, IEEE Trans. Pattern Anal. Mach. Intell. 23 (2001) 22–41.
[8] H. Cheng, K. Vu, K.A. Hua, Subspace projection: a unified framework for a class of partition-based dimension reduction techniques, Inform. Sci. 179 (2009) 1234–1248.
[9] T.F. Cox, M.A.A. Cox, Multidimensional Scaling, Chapman & Hall, London, 1994.
[10] D. De Ridder, R.P.W. Duin, Sammon mapping using neural networks: a comparison, Pattern Recogn. Lett. 18 (1997) 1307–1316.
[11] P. Demartines, J. Hérault, Curvilinear component analysis: a self-organizing neural network for nonlinear mapping of data sets, IEEE Trans. Neural Networks 8 (1997) 148–154.
[12] R. Durbin, G. Mitchison, A dimension reduction framework for understanding cortical maps, Nature 343 (1990) 644–647.
[13] P.A. Estévez, C.J. Figueroa, Online data visualization using the neural gas network, Neural Networks 19 (2006) 923–934.
[14] R.A. Fisher, The use of multiple measures in taxonomic problems, Ann. Eugenic. 7 (1936) 179–188.
[15] A. Flexer, Limitations of self-organizing maps for vector quantization and multidimensional scaling, Adv. Neural Inform. Process. Syst. 10 (1997) 445–451.
[16] R.T. Freeman, H. Yin, Adaptive topological tree structure for document organisation and visualisation, Neural Networks 17 (2004) 1255–1271.
[17] C. Fyfe, Two topographic maps for data visualisation, Data Min. Knowl. Disc. 14 (2007) 207–224.
[18] G.J. Goodhill, T. Sejnowski, A unifying objective function for topographic mappings, Neural Comput. 9 (1997) 1291–1303.
[19] A.N. Gorban, B. Kégl, D.C. Wunsch, A. Zinovyev, Principal Manifolds for Data Visualization and Dimension Reduction, Springer, 2008.
[20] S. Gunal, R. Edizkan, Subspace based feature selection for pattern recognition, Inform. Sci. 178 (2008) 3716–3726.
[21] J. Ham, D.D. Lee, S. Mika, B. Schölkopf, A kernel view of the dimensionality reduction of manifolds, in: Proceedings of 21st International Conference on Machine Learning, 2004, p. 47.
[22] T. Hastie, W. Stuetzle, Principal curves, J. Am. Stat. Assoc. 84 (1989) 502–516.
[23] X. He, D. Cai, P. Niyogi, Laplacian score for feature selection, Adv. Neural Inform. Process. Syst. 18 (2005).
[24] W. Huang, H. Yin, Linear and nonlinear dimensionality reduction for face recognition, Proceedings of International Conference on Image Processing, IEEE Press, Cairo, 2009. pp. 3337–3340.
[25] J. Karhunen, J. Joutsensalo, Generalisation of principal component analysis, optimisation problems, and neural networks, Neural Networks 8 (1995) 549–562.
[26] B. Kégl, A. Krzyzak, T. Linder, K. Zeger, A polygonal line algorithm for constructing principal curves, Adv. Neural Inform. Process. Syst. 11 (1998) 501–507.
[27] T. Kohonen, Self-organised formation of topologically correct feature map, Biol. Cybernet. 43 (1982) 56–69.
[28] T. Kohonen, Self-Organization and Associative Memory, Springer-Verlag, 1984.
[29] T. Kohonen, The adaptive-subspace SOM (ASSOM) and its use for the implementation of invariant feature detection, in: Proceedings of International Conference on Artificial Neural Networks, 1995, pp. 3–10.
[30] T. Kohonen, Self-Organising Maps, second ed., Springer, 1997.

[31] T. Korenius, J. Laurikkala, M. Juhola, On principal component analysis, cosine and Euclidean measures in information retrieval, Inform. Sci. 177 (2007) 4893–4995.

[32] M.A. Kramer, Nonlinear principal component analysis using autoassociative neural networks, AICHE (American Institute of Chemical Engineers) Journal 37 (1991) 233–243.

[33] K. Langus, S. Kaski, T. Kohonen, Mining massive document collections by the WEBSOM method, Inform. Sci. 163 (2004) 135–156.

[34] S. Lawrence, C.L. Giles, A.C. Tsoi, A.D. Back, Face recognition: a convolutional neural-network approach, IEEE Trans. Neural Networks 8 (1997) 98–113.

[35] M. LeBlanc, R.J. Tibshirani, Adaptive principal surfaces, J. Am. Stat. Assoc. 89 (1994) 53–64.

[36] R.C.T. Lee, J.R. Slagle, H. Blum, A triangulation method for the sequential mapping of points from n-space to two-space, IEEE Trans. Comput. 27 (1977) 288–292.

[37] D. Lowe, M.E. Tipping, Feed-forward neural networks and topographic mappings for exploratory data analysis, Neural Comput. Appl. 4 (1996) 83–95.

[38] S.P. Luttrell, Derivation of a class of training algorithms, IEEE Trans. Neural Networks 1 (1990) 229–232.

[39] S.P. Luttrell, A Bayesian analysis of self-organising maps, Neural Comput. 6 (1994) 767–794.

[40] E.C. Malthouse, Limitations of nonlinear PCA as performed with generic neural networks, IEEE Trans. Neural Networks 9 (1998) 165–173.

[41] J. Mao, A.K. Jain, Artificial neural networks for feature extraction and multivariate data projection, IEEE Trans. Neural Networks 6 (1995) 296–317.

[42] G. Mitchison, A type of duality between self-organizing maps and minimal wiring, Neural Comput. 7 (1995) 25–35.

[43] C.S. Möller-Levet, H. Yin, Modeling and analysis of gene expression time-series based on co-expression, Int. J. Neural Syst. 15 (2005) 311–322.

[44] D.V. Nguyen, D.M. Rockeb, On partial least squares dimension reduction for microarray-based classification: a simulation study, Comput. Stat. Data Anal. 46 (2004) 407–425.

[45] H. Ni, H. Yin, Self-organising mixture autoregressive model for non-stationary time series modelling, Int. J. Neural Syst. 18 (2008) 469–480.

[46] E. Oja, Neural networks, principal components, and subspaces, Int. J. Neural Syst. 1 (1989) 61–68.

[47] H. Ritter, T. Martinetz, K. Schulten, Neural Computation and Self-organising Maps: An Introduction, Addison-Wesley Publishing Company, 1992.

[48] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (2000) 2323–2326.

[49] J.W. Sammon, A nonlinear mapping for data structure analysis, IEEE Trans. Comput. 18 (1969) 401–409.

[50] T.D. Sanger, Optimal unsupervised learning in a single-layer linear feedforward network, Neural Networks 2 (1991) 459–473.

[51] B. Schölkopf, A. Smola, K.R. Müller, Nonlinear component analysis as a kernel eigenvalue problem, Neural Comput. 10 (1998) 1299–1319.

[52] F.Y. Shih, C.Y. Fu, K. Zhang, Multi-view face identification and pose estimation using B-spline interpolation, Inform. Sci. 169 (2005) 189–204.

[53] A.J. Smola, R.C. Williamson, S. Mika1, B. Schölkopf, Regularized principal manifolds, in: Proceedings of EuroCOLT'99, LNAI-1572, 1999, pp. 214–229.

[54] X. Tan, S. Chen, Z. Zhou, F. Zhang, Recognizing partially occluded, expression variant faces from single training image per person with SOM and soft *k*-NN ensemble, IEEE Trans. Neural Networks 16 (2005) 875–886.

[55] J.B. Tenenbaum, V. de Silva, J.C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (2000) 2319–2323.

[56] R. Tibshirani, Principal curves revisited, Stat. Comput. 2 (1992) 183–190.

[57] G. Wang, X. Zhou, B. Wang, B. Qiao, D. Han, A hyperplane based indexing technique for high-dimensional data, Inform. Sci. 177 (2008) 2255–2268.

[58] Y. Weiss, Segmentation using eigenvectors: a unified view, in: Proceedings of IEEE International Conference on Computer Vision, 1999, pp. 975–982.

[59] D.J. Willshaw, C. von der Malsburg, How patterned neural connections can be set up by self-organization, Proc. Roy. Soc. Lond. Ser. B 194 (1976) 431–445.

[60] S. Wu, T.W.S. Chow, PRSOM: a new visualization method by hybridizing multidimensional scaling and self-organizing map, IEEE Trans. Neural Networks 16 (2005) 1362–1380.

[61] H. Yin, Data visualisation and manifold mapping using the ViSOM, Neural Networks 15 (2002) 1005–1016.

[62] H. Yin, ViSOM – a novel method for multivariate data projection and structure visualisation, IEEE Trans. Neural Networks 13 (2002) 237–243.

[63] H. Yin, Resolution enhancement for the ViSOM, in: Proceedings of Workshop on Self-Organizing Maps, Kitakyushu, Japan, Kyushu Institute of Technology, 2003, pp. 208–212.

[64] H. Yin, Connection between self-organising maps and metric multidimensional scaling, in: Proceedings of International Joint Conference on Neural Networks, IEEE Press, Orlando, 2007, pp. 1025–1030.

[65] H. Yin, On multidimensional scaling and the embedding of self-organizing maps, Neural Networks 21 (2008) 160–169.

[66] H. Yin, The self-organizing maps: background, theories, extensions and applications, in: J. Fulcher, L.C. Jain (Eds.), Computational Intelligence: A Compendium, Springer, 2008, pp. 715–762.

[67] H. Yin, N.M. Allinson, On the distribution and convergence of the feature space in self-organising maps, Neural Comput. 7 (1995) 1178–1187.

[68] H. Yin, N.M. Allinson, Self-organising mixture networks for probability density estimation, IEEE Trans. Neural Networks 12 (2001) 405–411.

[69] H. Yin, P. Panzeri, Z. Mehboob, M. Diamond, Decoding population neuronal responses by topological clustering, in: Proceedings of International Conference on Artificial Neural Networks, LNCS-5164, 2008, pp. 547–556.