

Spherical Embedding and Classification^{*}

Richard C. Wilson and Edwin R. Hancock

Department of Computer Science
University of York
Heslington, York, UK
wilson@cs.york.ac.uk

Abstract. Most problems in pattern recognition can be posed in terms of using the dissimilarities between the set of objects of interest. A vector-space representation of the objects can be obtained by embedding them as points in Euclidean space. However many dissimilarities are non-Euclidean and cannot be represented accurately in Euclidean space. This can lead to a loss of information and poor performance. In this paper, we approach this problem by embedding the points in a non-Euclidean curved space, the hypersphere. This is a metric but non-Euclidean space which allows us to define a geometry and therefore construct geometric classifiers. We develop a optimisation-based procedure for embedding objects on hyperspherical manifolds from a given set of dissimilarities. We use the Lie group representation of the hypersphere and its associated Lie algebra to define the exponential map between the manifold and its local tangent space. We can then solve the optimisation problem locally in Euclidean space. This process is efficient enough to allow us to embed large datasets. We also define the nearest mean classifier on the manifold and give results for the embedding accuracy, the nearest mean classifier and the nearest-neighbor classifier on a variety of indefinite datasets.

1 Introduction

Many pattern recognition problems can be posed in terms of measuring the dissimilarities between a set of objects. This is a very general approach, as it is a superset of the classic feature-based approach. Nearly all approaches to recognition involve measuring a dissimilarity or distance and classifying on this basis. One approach to this problem is to embed objects into a vector-space using techniques such as multidimensional scaling or IsoMap[1]. Once embedded in such a space then the objects can be characterised by their embedding co-ordinate vectors, and analysed in a conventional manner using Euclidean distance.

There are however some limits to this paradigm; Euclidean distances are always *definite* and are intrinsically unable to represent dissimilarities which are *indefinite*. We discuss the issue of indefinite dissimilarities in more detail in the

^{*} We acknowledge financial support from the FET programme within the EU FP7, under the SIMBAD project (contract 213250). Edwin Hancock was also supported by a Royal Society Wolfson Research Merit Award.

next section. In practice, many dissimilarity measures are indefinite; examples include shape-similarities, and distance measures used in gesture interpretation and graph comparison, but there are many more. Any method of comparison which relies on local alignment or variable local control parameters has the potential to produce indefinite (non-Euclidean) dissimilarities.

One alternative is to ‘correct’ the data to remove the indefinite part. However, previous work[2] has shown that there is potentially useful information in the non-Euclidean part of the dissimilarities, and removing this can result in worse performance. Another alternative is to embed the data in a pseudo-Euclidean space, i.e. one where certain dimensions are characterised by negative eigenvalues and the squared-distance between objects has positive and negative components which sum together to give the total distance. A pseudo-Euclidean space is however non-metric, which makes it difficult to correctly compute the geometric quantities required by many classifiers. This is because locality is not preserved in this space; two points which are far apart can both be close to a third point.

A third alternative, which we explore here, is to use a non-Euclidean, but metric, embedding space. A Riemannian manifold is curved, and the geodesic distances are metric. However they can also be indefinite and so can represent indefinite dissimilarities. In this paper, we explore the embedding of objects onto the hypersphere with its associated spherical geometry. Non-Euclidean embeddings have been reported elsewhere in the literature. For example, Lindman and Caelli have studied both spherical and hyperbolic embeddings in the context of interpreting psychological data[3]. Cox and Cox[4] describe multidimensional scaling constrained to a spherical space and optimise the stress to find a good embedding. Shavitt and Tankel have used the hyperbolic embedding as a model of internet connectivity[5]. Hubert et al have investigated the use of unidimensional embeddings on circles[6]. Robles-Kelly and Hancock[7] preprocess the available similarity data so that it conforms either to elliptic or hyperbolic geometry. In practice the former corresponds to a scaling of the distance using a sine function, and the latter scaling the data using a hyperbolic sine function.

In this paper, we propose a optimisation-based procedure for embedding objects on hyperspherical manifolds. The purpose of this embedding is to faithfully represent the dissimilarities between objects in a metric space. A metric space is important because it allows us to compute statistics and define geometric constructs such as boundaries, in contrast to a non-metric space where non-locality is a problem. We also define the nearest mean classifier on the manifold and give results for the embedding accuracy, the nearest mean classifier and the nearest-neighbor classifier on a variety of indefinite datasets. The optimisation approach we use employs the Lie group representation of the hypersphere and its associated Lie algebra to define the exponential map between the manifold and its local tangent space. We can then solve the optimisation problem locally in Euclidean space. This process is efficient enough to allow us to embed datasets of several thousand objects.

2 Indefinite Spaces

We begin with the assumption that we have a set of objects of interest and have measured a set of dissimilarities or distances between all pairs of objects in our problem. This is denoted by the matrix \mathbf{D} , where D_{ij} is the distance between objects i and j . We can define an equivalent set of similarities by using the matrix of squared distances \mathbf{D}' , where $D'_{ij} = D_{ij}^2$. This is achieved by identifying the similarities as $-\frac{1}{2}\mathbf{D}'$ and centering the resulting matrix:

$$\mathbf{S} = -\frac{1}{2}\left(\mathbf{I} - \frac{1}{n}\mathbf{J}\right)\mathbf{D}'\left(\mathbf{I} - \frac{1}{n}\mathbf{J}\right) \tag{1}$$

Here \mathbf{J} is the matrix of all-ones, and n is the number of objects. In Euclidean space, this procedure gives exactly the inner-product or kernel matrix for the points.

If \mathbf{S} is positive semi-definite, then the original distances are Euclidean and we can use the kernel embedding to locate positions \mathbf{x}_i for the points in Euclidean space as follows

$$\mathbf{S} = \mathbf{U}_S \mathbf{\Lambda}_S \mathbf{U}_S^T = \mathbf{X}\mathbf{X}^T \tag{2}$$

$$\mathbf{X} = \mathbf{U}_S \mathbf{\Lambda}_S^{\frac{1}{2}} \tag{3}$$

where \mathbf{U}_S and $\mathbf{\Lambda}_S$ are the eigenvector and eigenvalue matrices of \mathbf{S} , respectively. The position-vector \mathbf{x}_i of the i^{th} point corresponds to the i^{th} row of \mathbf{X} .

If \mathbf{S} is indefinite, which is often the case, then the objects cannot exist in Euclidean space with the given distances. This does not necessarily mean the the distances are non-metric; metricity is a separate issue. One measure of the deviation from definiteness which has proved useful is the negative eigenfraction (NEF) which measures the fractional weight of eigenvalues which are negative[8]:

$$\text{NEF} = \frac{\sum_{\lambda_i < 0} |\lambda_i|}{\sum_i |\lambda_i|} \tag{4}$$

If NEF=0, then the data is definite and can be represented by points in Euclidean space. We can measure the *non-metricity* of the data by counting the number of violations of metric properties. It is very rare to have an initial distance measure which gives negative distance, so we will assume than the distances are all positive. The two measures of interest are then the fraction of triples which violate the triangle inequality (TV) and the degree of asymmetry of the distances (γ)[2]:

$$\gamma = \sum_{i \neq j} \frac{|\tilde{d}(i, j) - \tilde{d}(j, i)|}{|\tilde{d}(i, j) + \tilde{d}(j, i)|} \tag{5}$$

where $\tilde{d}(., .)$ is the dissimilarity scaled so that the average dissimilarity is one.

If the data is metric (or, in practice, close to metric) but indefinite then we must use a curved space to embed the points.

3 Spherical Space

A spherical space is an example of a Riemannian manifold. On the manifold, distances are measured by geodesics (the shortest curve between points), and geodesic distances are metric. Spherical space is curved however, and so the distances are fundamentally non-Euclidean and in general the similarity matrix of points in spherical space will be indefinite. This makes it a potential choice for representing non-Euclidean datasets.

A manifold embedding is important because it allows the use of geometric and statistical tools on the embedded points. On a Riemannian manifold, distances are defined between any pair of points in the manifold in a consistent way (not just between the sample data-points). Geodesic distance is defined as the length of the shortest curve which joins two points (the curve is known as a geodesic), and is a metric. Geodesics are the equivalent of straight lines in Euclidean space, and allow us to construct a geometry in curved space. We can also compute statistics such as the mean in a way consistent with the normal Euclidean definition. This means that all the standard classifiers can be applied (at least in theory) to the data, but the exact formulation will differ from vector-space classifiers.

The spherical manifold in 2D is isomorphic to the 2D surface of a sphere embedding in 3D space, which has a well-known parametric form. Here r is the radius of the sphere, u is the azimuth angle and v is the zenith angle.

$$\mathbf{x} = (r \sin u \sin v, r \cos u \sin v, r \cos v)^T \quad (6)$$

This geometry generalises to an $n - 1$ dimensional hypersphere embedded in an n -dimensional Euclidean space. The surface can be defined implicitly using the constraint

$$\sum_i x_i^2 = r^2 \quad (7)$$

where r is the radius of the hypersphere. This surface is curved and has a constant sectional curvature of $K = 1/r^2$ everywhere.

The geodesic distance between two points in curved space is the length of the shortest curve lying in the space and joining the two points. On the hypersphere, the geodesic is a great circle. The distance is the length of the arc of the great circle which joins the two points. If the angle subtended by two points at the centre of the hypersphere is θ_{ij} , then the distance between them is

$$d_{ij} = r\theta_{ij} \quad (8)$$

With the coordinate origin at the centre of the hypersphere, we can represent a point by a position vector \mathbf{x}_i of length r . Since the inner product is $\langle \mathbf{x}_i, \mathbf{x}_j \rangle = r^2 \cos \theta_{ij}$ we can also write

$$d_{ij} = r \cos^{-1} \frac{\langle \mathbf{x}_i, \mathbf{x}_j \rangle}{r^2} \quad (9)$$

4 The Exponential Map

Our procedure for embedding points on a sphere requires one important tool of Riemannian geometry, which is the exponential map. The exponential map is a map from points on the manifold to points on a tangent space of the manifold. As the tangent space is flat (i.e. Euclidean), we can calculate quantities in a straightforward way. The map has an origin, which defines the point at which we construct the tangent space of the manifold. The formal definition of the Exponential map is the map which connects the Lie algebra on the tangent space to the Lie group which defines the manifold. We will not concern ourselves with the technical details here, but the map has an important property which simplifies geometric computations; the geodesic distance between the origin of the map and a point on the manifold is the same as the Euclidean distance between the images of the two points on the tangent space. Formally, the definition of these properties as follows: Let T_M be the tangent space at some point M on the manifold, P be a point on the manifold and X a point on the tangent space. We have

$$X = \text{Log}_M P \quad (10)$$

$$P = \text{Exp}_M X \quad (11)$$

$$d_g(P, M) = d_e(X, M) \quad (12)$$

The Log and Exp notation defines a log-map from the manifold to the tangent space and an exp-map from the tangent space to the manifold. This is a formal notation and does not imply the normal log and exp functions - although they do co-incide for some types of data, they are not the same for the spherical space. M is the origin of the map and is mapped onto the origin of the tangent space. The distance $d_g(.,.)$ is the geodesic distance on the manifold and $d_e(.,.)$ the Euclidean distance on the tangent space.

For the spherical manifold, the exponential map is as follows. We define a point P on our manifold as a position vector \mathbf{p} with length r (the origin is at the centre of the hypersphere). Similarly, the point M is represented by the vector \mathbf{m} , and M is the origin of the map. The maps are then

$$\mathbf{x} = \frac{\theta}{\sin \theta} (\mathbf{p} - \mathbf{m} \cos \theta) \quad (13)$$

$$\mathbf{p} = \mathbf{m} \cos \theta + \frac{\sin \theta}{\theta} \mathbf{x} \quad (14)$$

$$d_g(P, M) = d_e(X, M) = |\mathbf{x}| = r\theta \quad (15)$$

where $\theta = \cos^{-1} \langle \mathbf{p}, \mathbf{m} \rangle / r^2$. The vector \mathbf{x} is the image of P in the tangent space, and the image of M is at the origin of the tangent space.

5 Spherical Embedding

Given a dissimilarity matrix \mathbf{D} , we want to find the embedding of a set of points on the surface of a hypersphere of radius r , such that the geodesic distances are

as similar as possible to \mathbf{D} . Unfortunately, this appears to be a hard problem and therefore we use an approximate optimisation-based approach. We simplify the problem by considering just the distances to a single point at a time. Let the point of interest be \mathbf{p}_i ; we then want to find a new position for this point on the hypersphere such that the geodesic distance to point j is d_{ij}^* where $*$ denotes that this is the target distance. We formulate the estimation of position as a least-squares problem which minimises

$$E = \sum_{j \neq i} (d_{ij}^2 - d_{ij}^{*2})^2 \tag{16}$$

where d_{ij} is the actual distance between the points. This is a similar formulation to Cox and Cox[4] and other approaches to non-Euclidean multidimensional scaling, who seek to minimise the ‘stress’. Direct optimisation on the sphere is complicated by the need to restrict points to the manifold. However, as we are considering a single point at a time, we can construct a linear embedding using the log-map and optimise in the Euclidean space. This is a different approach to that of Cox and Cox[4]. If the current point-positions on the hypersphere are $\mathbf{p}_j, \forall j$, we can use the log-map to obtain point-positions for each object in the tangent space of $\mathbf{x}_j, \forall j$ as follows:

$$\mathbf{x}_j = \text{Log}_{\mathbf{p}_i} \mathbf{p}_j = \frac{\theta_{ij}}{\sin \theta_{ij}} (\mathbf{p}_j - \mathbf{p}_i \cos \theta_{ij}) \tag{17}$$

with $\mathbf{x}_i = 0$.

We have found standard optimisation schemes to be infeasible on larger datasets, so here we propose a gradient descent scheme with optimal step-size. In this iterative scheme, we update the position of the point \mathbf{x}_i in the tangent space to obtain a better fit to the given distances. At iteration k , the point is at position $\mathbf{x}_i^{(k)}$. Initially, the point is at the origin, so $\mathbf{x}_i^{(0)} = 0$. Since the points lie in tangent space, which is Euclidean, we then have $d_{ij}^2 = (\mathbf{x}_j - \mathbf{x}_i)^T (\mathbf{x}_j - \mathbf{x}_i)$ and gradient of the error is

$$\nabla E = 4 \sum_{j \neq i} (d_{ij}^2 - d_{ij}^{*2}) (\mathbf{x}_i - \mathbf{x}_j) \tag{18}$$

and our iterative update procedure is

$$\mathbf{x}_i^{(k+1)} = \mathbf{x}_i^{(k)} + \eta \nabla E \tag{19}$$

Finally, we can determine the optimal step size as follows: let $\Delta_j = d_{ij}^2 - d_{ij}^{*2}$ and $\alpha_j = \nabla E^T (\mathbf{x}_i - \mathbf{x}_j)$, then the optimal step size is the smallest root of the cubic

$$n |\nabla E|^2 \eta^3 + 3 |\nabla E|^2 \left(\sum_j \alpha_j \right) \eta^2 + \left(2 \sum_j \alpha_j^2 + |\nabla E|^2 \sum_j \Delta_j \right) \eta + \sum_j \alpha_j \Delta_j \tag{20}$$

This step-size is optimal in the sense that it minimises the error in the direction of the gradient.

After finding a new point position \mathbf{x}_i , we apply the exp-map to locate the new point position on the spherical manifold

$$\mathbf{p}'_i = \mathbf{p}_i \cos \theta + \frac{\sin \theta}{\theta} \mathbf{x}_i \quad (21)$$

5.1 Classifiers in the Manifold

As well as embedding distances on the spherical manifold, it is important to be able to perform operations such as classification in the manifold. Some classifiers are trivially implemented on a spherical manifold, for example the nearest-neighbors(NN). Others which utilise geometry must be modified to incorporate the non-Euclidean geometry of curved space. Here we discuss the nearest mean classifier(NMC) in a non-flat manifold.

The *intrinsic mean* of a set of points on the manifold may be computed via the generalised mean[9]

$$\bar{P} = \arg \min_P \sum_i d_g(P, P_i) \quad (22)$$

We can solve for the mean of a set of points in a manifold using the following iterative procedure involving the exponential map[9]:

$$\mathbf{m}^{(k+1)} = \text{Exp}_{\mathbf{m}^{(k)}} \frac{1}{n} \sum_i \text{Log}_{\mathbf{m}^{(k)}} \mathbf{p}_i \quad (23)$$

While the convergence of this process is not guaranteed in a general manifold, it is well behaved on the hypersphere[9]. As a result, we can compute the means of each class $\mathbf{m}_1, \dots, \mathbf{m}_C$ and implement the NMC:

$$c^* = \arg \min_c \left[r \cos^{-1} \frac{\langle \mathbf{x}, \mathbf{m}_c \rangle}{r^2} \right] \quad (24)$$

6 Experimental Results

We have applied our embedding method to a number of indefinite datasets. These are summarised in the table below, along with their degree of indefiniteness, as measured by the negative eigenfraction (Eqn. 4). These datasets are produced by dissimilarity measures applied to a variety of real world problems. The Coil datasets are produced by graph-matching algorithms applied to corner-graphs of some of the objects in the COIL database[10], using graduated assignment[11](CoilYork) and the JoEig approach[12](CoilDelftDiff and CoilDelftSame). The CatCortex data gives the similarity between different cortical regions in terms of connectivity[13]. The DelftGestures dataset consists of the dissimilarities computed from a set of gestures in a sign-language using a dynamic time warping procedure[14]. The FlowCyto series of datasets is based

Table 1. Properties of datasets used

Dataset	Size	NEF	Triangle violations	Asymmetry
CoilYork	288	0.258	1/23639616	0.009
DelftGestures	1500	0.308	14798/3368253000	0
FlowCyto-1	612	0.275	272052/228098520	0
FlowCyto-2	612	0.268	161517/228098520	0
FlowCyto-3	612	0.275	272879/228098520	0
FlowCyto-4	612	0.272	268991/228098520	0
Newsgroups	600	0.202	4643/214921200	0
Chickenpieces-5	446	0.216	0/88120680	0.044
Chickenpieces-10	446	0.257	1/88120680	0.046
Chickenpieces-15	446	0.286	74/88120680	0.051
Chickenpieces-20	446	0.307	695/88120680	0.057
Chickenpieces-25	446	0.320	1375/88120680	0.063
Chickenpieces-30	446	0.331	3188/88120680	0.067
Chickenpieces-35	446	0.339	4834/88120680	0.073
Chickenpieces-40	446	0.345	7549/88120680	0.076
CatCortex	65	0.272	286/262080	0
CoilDelftDiff	288	0.128	1/23639616	0
CoilDelftSame	288	0.027	0/23639616	0
WoodyPlants50	791	0.229	115253/493038210	0
ProDom	2604	0.043	136/17636907624	0
Zongker	2000	0.419	6583656/7988004000	0.051

on the L1-norm dissimilarities between flowcytometer histograms of breast cancer tissues. The data were acquired by M. Nap and N. van Rodijnen of the Atrium Medical Center in Heerlen, The Netherlands, during 2000-2004. Newsgroups is a small subset of the 20Newsgroups data of Roweis. The ProDom dataset is a set of dissimilarities derived from the structural matching of protein domain sequences[15]. WoodyPlants50 is a dataset of shape dissimilarities between plant leaves[16]. The Zongker dissimilarities are based on deformable template matching between 2000 handwritten digits in 10 classes[17]. Finally, the Chickenpiece dataset is another set of shape dissimilarities derived from string-edit distance on the contours of chicken piece silhouettes[2]. This data has a number of controllable parameters which influence the indefinite nature of the dissimilarities. Here we use and edit cost of 45 and a variety of contour lengths (5,10,15,20,25,30,35,40).

We characterise the accuracy of our embeddings in two different ways. Firstly we measure the RMS fractional error of the embedded distances:

$$\text{RMS Error} = \left(\frac{1}{n} \sum_{ij} \frac{D_{ij} - D_{ij}^*}{\bar{D}} \right) \quad (25)$$

where \bar{D} is the average dissimilarity between objects in the original data. Secondly, we measure the 1NN classifier error, both before and after embedding.

Table 2. Embedding results for the datasets (in order of increasing error)

Dataset	Size	1NN (orig)	Error	Radius	1NN (emb)	NMC
Newsgrroups	600	0.269 ± 0.015	0.022	0.6298	0.279 ± 0.012	0.208 ± 0.015
CoilDelftDiff	288	0.487 ± 0.033	0.030	0.0277	0.479 ± 0.022	0.467 ± 0.034
Chickenpieces-5	446	0.350 ± 0.022	0.030	66.9	0.417 ± 0.022	0.407 ± 0.02
WoodyPlants50	791	0.101 ± 0.008	0.034	0.4362	0.147 ± 0.015	0.197 ± 0.016
Chickenpieces-10	446	0.170 ± 0.016	0.039	33.4	0.249 ± 0.018	0.338 ± 0.022
DelftGestures	1500	0.042 ± 0.0048	0.039	3.9826	0.135 ± 0.009	0.104 ± 0.004
Chickenpieces-15	446	0.079 ± 0.011	0.049	20.73	0.116 ± 0.018	0.249 ± 0.028
Chickenpieces-20	446	0.069 ± 0.012	0.052	17	0.109 ± 0.011	0.202 ± 0.022
Chickenpieces-25	446	0.048 ± 0.01	0.057	13.1	0.086 ± 0.013	0.21 ± 0.025
FlowCyto-2	612	0.366 ± 0.019	0.059	12132	0.378 ± 0.017	0.389 ± 0.028
Chickenpieces-30	446	0.048 ± 0.009	0.062	11.01	0.091 ± 0.013	0.197 ± 0.015
CoilYork	288	0.278 ± 0.025	0.063	177.8	0.307 ± 0.024	0.471 ± 0.029
FlowCyto-3	612	0.413 ± 0.013	0.072	13078	0.421 ± 0.021	0.4 ± 0.015
Chickenpieces-35	446	0.065 ± 0.011	0.073	10.12	0.069 ± 0.007	0.178 ± 0.023
Chickenpieces-40	446	0.087 ± 0.014	0.078	8.14	0.099 ± 0.012	0.2 ± 0.015
FlowCyto-1	612	0.369 ± 0.013	0.078	12794	0.425 ± 0.008	0.385 ± 0.02
CatCortex	65	0.095 ± 0.034	0.084	2.33	0.111 ± 0.04	0.047 ± 0.025
FlowCyto-4	612	0.425 ± 0.023	0.090	11761	0.413 ± 0.018	0.436 ± 0.026
ProDom	2604	0.002 ± 0.001	0.122	471.1	0.038 ± 0.003	0.21 ± 0.011
CoilDelftSame	288	0.636 ± 0.031	0.134	0.0577	0.674 ± 0.040	0.433 ± 0.038
Zongker	2000	0.372 ± 0.016	0.233	0.2887	0.043 ± 0.005	0.109 ± 0.009

This demonstrates whether the embedding preserves the local structure of the classes adequately. In the final column we show the performance of the NMC classifier on the hypersphere.

The results show that we obtain an accuracy spherical embedding for nearly all the data. Of the 21 datasets, only three have more than 10% RMS error on the embedding. This demonstrates the effectiveness of our embedding technique at locating optimal embeddings. For ten of the datasets, we see virtually identical 1NN performance both before and after embedding, and for one a large improvement (Zongker). We do not know the cause of this unexpected behaviour, but it seems to be a feature of this particular dataset. For the other ten sets, we see deterioration in the 1NN classification, indicating that the local structure has been changed somewhat. This is particularly evident in the Chickenpieces data, for which six of the eight examples give worse 1NN scores. It seems that this data series is unsuitable for spherical embedding.

The NMC classifier shows a far wider range of performance. The Chickenpieces data series, CoilYork, WoodyPlants50 and ProDom show a substantially worse performance with the NMC than with the original 1NN classifier, whereas Newsgrroups, CatCortex, CoilDelftSame and Zongker show a substantial improvement.

7 Conclusions

In this paper we used spherical embedding as a solution to the problem of indefinite, non-Euclidean dissimilarities. This embedding preserves some of the non-Euclidean nature of the dissimilarities which may be important in other tasks such as classification. We developed an optimisation-based procedure for embedding objects on hyperspherical manifolds which uses the Lie group representation of the hypersphere and its associated Lie algebra to define the exponential map between the manifold and its local tangent space. The optimisation is then solved locally in Euclidean space. This process is efficient enough to allow us to embed datasets of several thousand objects. We also defined the nearest mean classifier on the manifold.

Experiments on a variety of non-Euclidean datasets show that we can obtain accurate embeddings representing the dissimilarities on the hypersphere. The classification results show that the embedding of some datasets is very useful (for example the Newsgroups data), and for others not effective (the Chickens data).

References

1. Tenenbaum, J.B., de Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. *Science* 290, 2319–2323 (2000)
2. Pekalska, E., Harol, A., Duin, R.P.W., Spillmann, B., Bunke, H.: Non-euclidean or non-metric measures can be informative. In: SSPR/SPR, pp. 871–880 (2006)
3. Lindman, H., Caelli, T.: Constant curvature riemannian scaling. *Journal of Mathematical Psychology* 17, 89–109 (1978)
4. Cox, T.F., Cox, M.A.A.: Multidimensional scaling on a sphere. *Communications in Statistics - Theory and Methods* 20(9), 2943–2953 (1991)
5. Shavitt, Y., Tankel, T.: Hyperbolic embedding of internet graph for distance estimation and overlay construction. *IEEE/ACM Transactions on Networking* 16, 25–36 (2008)
6. Hubert, L., Arabie, P., Meulman, J.: Linear and circular unidimensional scaling for symmetric proximity matrices. *British Journal of Mathematical & Statistical Psychology* 50, 253–284 (1997)
7. Robles-Kelly, A., Hancock, E.R.: A riemannian approach to graph embedding. *Pattern Recognition* 40, 1042–1056 (2007)
8. Pekalska, E., Duin, R.P.W.: *The dissimilarity representation for pattern recognition*. World Scientific, Singapore (2005)
9. Fletcher, P.T., Lu, C., Pizer, S.M., Joshi, S.: Principal geodesic analysis for the study of nonlinear statistics of shape. *IEEE Transactions on Medical Imaging* 23(8), 995–1005 (2004)
10. Xiao, B., Hancock, E.R.: Geometric characterisation of graphs. In: Roli, F., Vitulano, S. (eds.) *ICIAP 2005*. LNCS, vol. 3617, pp. 471–478. Springer, Heidelberg (2005)
11. Gold, S., Rangarajan, A.: A graduated assignment algorithm for graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 18, 377–388 (1996)

12. Lee, W.J., Duin, R.P.W.: An inexact graph comparison approach in joint eigenspace. In: da Vitoria Lobo, N., Kasparis, T., Roli, F., Kwok, J.T., Georgiopoulos, M., Anagnostopoulos, G.C., Loog, M. (eds.) S+SSPR 2008. LNCS, vol. 5342, pp. 35–44. Springer, Heidelberg (2008)
13. Scannell, J., Blakemore, C., Young, M.: Analysis of connectivity in the cat cerebral cortex. *Journal of Neuroscience* 15, 1463–1483 (1995)
14. Lichtenauer, J., Hendriks, E.A., Reinders, M.J.T.: Sign language recognition by combining statistical dtw and independent classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30, 2040–2046 (2008)
15. Roth, V., Laub, J., Buhmann, J., Mueller, K.R.: Going metric: Denoising pairwise data. *Advances in Neural Information Processing Systems*, 841–856 (2003)
16. Ling, H., Jacobs, D.: Shape classification using the inner-distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29, 286–299 (2007)
17. Jain, A., Zongker, D.: Representation and recognition of handwritten digits using deformable templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19, 1386–1391 (1997)