H-MDS: A new approach for Interactive Visualization with Multidimensional Scaling in the Hyperbolic Space

Jörg A. Walter

Neuroinformatics Group · Department of Computer Science
University of Bielefeld · 33615 Bielefeld · Germany

Abstract

We introduce a novel projection based visualization method for high-dimensional datasets by combining concepts from MDS and the geometry of the hyperbolic spaces. This approach *Hyperbolic Multi-Dimensional Scaling* (H-MDS) is a synthesis of two important concepts for explorative data analysis and visualization:

(i) Multi-dimensional scaling uses proximity or pair distance data to generate a low-dimensional, spatial presentation of the data. *(ii)* Previous work on the "hyperbolic tree browser" demonstrated the extraordinary advantages for an interactive display of graph-like data in the two-dimensional hyperbolic space ($\mathbb{H}^2$).

In the new approach, H-MDS maps proximity data directly into the $\mathbb{H}^2$. This removes the restriction to "quasi-hierarchical", graph-based data – a major limitation of *(ii)*. Since a suitable distance function can convert all kinds of data to proximity (or distance-based) data this type of data can be considered the most general.

We review important properties of the hyperbolic space and in particular the circular Poincaré model of the $\mathbb{H}^2$. It enables effective human-computer interaction: by mouse dragging the "focus", the user can navigate in the data without loosing the context. In $\mathbb{H}^2$ the "fish-eye" behavior originates not simply by a non-linear view transformation but rather by extraordinary, non-Euclidean properties of the $\mathbb{H}^2$. Especially, the exponential growth of length and area of the underlying space makes the $\mathbb{H}^2$ a prime target for mapping hierarchical and (now also) high-dimensional data.

Several high-dimensional mapping examples including synthetic and real world data are presented. Since high-dimensional data produce "ring" shaped displays, we present methods to enhance the display by modulating the dissimilarity contrast. This is demonstrated for an application for unstructured text: i.e., by using multiple film critiques from news:rec.art.movies.reviews and www.imdb.com, each movie is placed within the $\mathbb{H}^2$ – creating a "space of movies" for interactive exploration.

Key words: visualizing high-dimensional data, hyperbolic multi-dimensional scaling, H-MDS, focus+context, semantic browsing, text mining.
PACS: xxx

1 Introduction

For many tasks of exploratory data analysis visualization plays an important role. It is a key for efficient integration of human expertise – not only to include his background knowledge, intuition and creativity, but also his powerful pattern recognition and processing capabilities. The design goals for an optimal user interaction strongly depend on the given visualization task but they certainly include an easy and intuitive navigation with strong support for the user's orientation.

Since most of available data display devices are two-dimensional – paper and screens – the following problem must be solved: finding a meaningful spatial mapping of data onto the display area. One limiting factor is the "restricted neighborhood" around a point in a Euclidean 2D surface. *Hyperbolic spaces* open an interesting loophole. The extraordinary property of exponential growth of neighborhood with increasing radius – around all points – enables one to build novel displays.

The *"hyperbolic tree viewer"*, developed at Xerox Parc [7], demonstrates the remarkable elegant interactive capabilities (visit http://www.inxight.com for a Java demo). The hyperbolic model appears as a continuously graded, focus+context mapping to the display. Very quickly the navigation by mouse clicks and drags appear natural and intuitive. It supports up to 10 times as many nodes as conventional approaches while providing more effective navigation around the tree-hierarchy.

The question how effective is visualization and navigation in the hyperbolic space was studied by Pirolli et al. [12]. By conducting eye-tracker experiments they found that the focus+context navigation can significantly accelerate the "information foraging". Risken et al. [14] compared traditional and hyperbolic browsers and found significant improvement in task time for this novel display type.

Unfortunately, previous usage of *direct* hyperbolic visualization was limited to hierarchical, tree-like, or "quasi-graph" data. A $\mathbb{H}^2$ grid layout of data was recently introduced by Helge Ritter by generalizing Kohonen's Self-Organizing Map algorithm (HSOM) to the hyperbolic space [15]. Jörg Ontrup applied this HSOM successfully to text categorization and browsing [11].

In this work we go one step further and introduce the *hyperbolic multi-dimensional scaling* (H-MDS) for a direct construction of a distance preserving embedding

Email address: walter@techfak.uni-bielefeld.de (Jörg A. Walter).

[1] This is an extended version of a talk [20] given on the ACM SIGMOD KDD'02 conference in Edmonton, Canada August 2002

of high-dimensional data into the hyperbolic space. The resulting H-MDS combines a number of attractive features of multi-dimensional scaling and hyperbolic spaces for an interactive display. Compared to the HSOM it is not based on a discrete grid and can be used also in situations where only dissimilarity data but not a vector representation is available.

In Sec. 2 and 3 we introduce the hyperbolic space and the standard multi-dimensional scaling. Sec. 4 explains the combination of both concepts into H-MDS. Even though the look and feel of an interactive visualization and navigation is hardly compressible to paper format, we report several results and snapshots of first experiments in Sec. 5. The effect of rescaling the dissimilarities is explored. In Sec. 6 we apply the H-MDS for spanning a space of unstructured text documents, i.e., special "averaged" film critiques. This approach allows to navigate in the space of selected movies. Since this dataset comprises many dimensions, the visual appearance as rings is discussed and methods to modulate the dissimilarity contrast are presented.

2 The Hyperbolic Space $\mathbb{H}^2$

2300 years ago, the Greek mathematician Euclid founded his geometry on five axioms. The fifth, the "parallel axiom", appeared unnecessary to many of his colleagues. And they tried hard to prove it derivable – without success. After almost 2000 years Lobachevsky (1793-1856), Bolyai (1802-1860), and Gauss (1777-1855) negated the axiom and independently discovered the non-Euclidean geometry. There exist only two geometries with constant non-zero curvature. Through our sight of common spherical surfaces (e.g. earth, orange) we are familiar with the *spherical geometry* and its constant *positive* curvature. Its counterpart with constant *negative* curvature is known as the *hyperbolic plane* $\mathbb{H}^2$ (with analogous generalizations to higher dimensions) [3,18]. Unfortunately, there is no "good" embedding of the $\mathbb{H}^2$ in $\mathbb{R}^3$, which makes it harder to grasp the unusual properties of the $\mathbb{H}^2$. Local patches resemble the situation at a saddle point, where the neighborhood grows faster than in flat space (see Fig. 1). Standard textbooks on Riemannian geometry (see, e.g. [8]) examine the relationship and expose that the circumference c and area a of a circle of hyperbolic radius ρ are given by

$$\text{area:} \quad a(\rho) = 4\pi \sinh^2(\rho/2) \tag{1}$$

$$\text{circumference:} \quad c(\rho) = 2\pi \sinh(\rho) \ . \tag{2}$$

This bears two remarkable asymptotic properties, *(i)* for small radius ρ the space "looks flat" since $a(\rho) \approx \pi\rho^2$ and $c(r) \approx 2\pi\rho$. *(ii)* For larger ρ both grow *exponentially* with the radius. As observed in [7,6], this trait makes the

hyperbolic space ideal for embedding hierarchical structures. Fig. 1 illustrates the spatial relations by embedding a small patch of the $\mathbb{H}^2$ in $\mathbb{R}^3$.

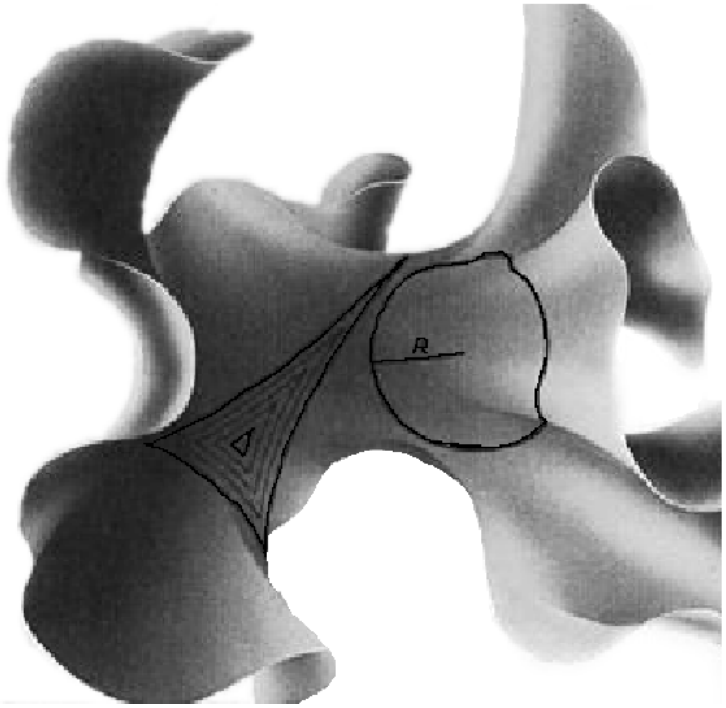

Fig. 1. There is literally more room in hyperbolic space than in Euclidean space, as shown in this illustrated embedding of the hyperbolic plane into 3D Euclidean space (from [19], courtesy of Jeffrey Weeks). *(Right:)* Exponential growth (Eq. 1) of the circumference $c(r)$ and area $a(r)$ is experienced if a "circle" with radius r is drawn in the wrinkling structure. *(Left:)* The sum of angles in a triangle is smaller than 180°. A $\mathbb{H}^2$ paper model of the $\mathbb{H}^2$ can be made by gluing many equilateral triangles along their edges in such a way that seven triangles meet at each vertex. Fig. 3 shows the Poincaré projection for eight meeting triangles.

To use the visualization potential of the $\mathbb{H}^2$ we must solve two problems: *(i)* the data must be "accommodated" in the hyperbolic space (see Sec. 4) and *(ii)* in order to inspect the result we need a projection of the $\mathbb{H}^2$ onto a suitable display. For practical and technological reasons this is a "flat surface" (we will not be able to buy a hyperbolic screen – at least for the foreseeable future). Fortunately, the projection problem was solved more than a century ago.

2.1 Projections of the Hyperbolic Space $\mathbb{H}^2$

It lays in the nature of a curved space to resist the *perfect* projection into the flat Euclidean surface. Each attempt compromises one or more correct representations of length, area, and angle (form) relations, as is well known from the spherical geometry (e.g., the Mercator, Lambert, and stereographic

projection). Similarly to the spherical geometry several mappings were developed, four are especially well examined: *(i)* the *Minkowski*, *(ii)* the *upper-half plane*, *(iii)* the *Klein-Beltrami*, and *(iv)* the *Poincaré* or *disk* mapping. See [3] for more details and geometric mapping to convert in-between *(i)–(iv)*.

2.2 Properties of the Poincaré Projection

What are its properties and what makes the Poincaré projection for our purpose the most suitable?

Display compatibility: The infinite large area of the $\mathbb{H}^2$ is mapped entirely into a circle, the Poincaré disk *PD*. This infinity representation fascinated M. Escher and inspired him to Fig. 2.

Circle rim "$= \infty$" : All remote points are close to the rim, without touching it.

Focus+Context: The *focus* can be moved to each location in $\mathbb{H}^2$, like a *"fovea"*. The zooming factor is 0.5 in the center and falls (exponentially) off with distance to the fovea. Therefore, the context appears very natural. As more remote things are, the less spatial representation is assigned in the current display.

Lines become circles: All $\mathbb{H}^2$-lines[2] appear as *circle* arc segments or (centered) straight lines in *PD*. There extensions cross the *PD*-rim always perpendicular on both ends[3], see Fig. 3.

Conformal mapping: Angles (and therefore form) relations are preserved in *PD*, area and length relations obviously not (in contrast, e.g., to the Klein-Beltrami model which is length preserving).

Parallel axiom can be demonstrated. While the Euclidean space accommodates exactly one parallel[4] to a line through a given point (not laying on this line), the $\mathbb{H}^2$ offers infinitely many. In *PD* a line is an arc who's ends are perpendicular to the circle rim. It is easy to draw many non-crossing circles through any given isolated point (see Fig. 3).

Exponential large space: There exist two kinds of "parallels": *(i) asymptotic parallels*, circles which touch at the rim in the same "∞-point", and *(ii) ultra parallels*, circle arcs which do not intersect within *PD*. One can anticipate, there is much more space in ∞ than "usual" – sometimes the $\mathbb{H}^2$ is also called "more intensive infinite" than the $\mathbb{R}^2$. This extra space is desired for finding good accommodation for our data.

---

[2] A *line* is by definition the shortest path between two points

[3] Sometimes straight lines through the center point are referred to as *generalized circle* and considered as a circle arc with infinite radius.

[4] A *parallel* in a two dimensional manifold is a line which does not intersect.

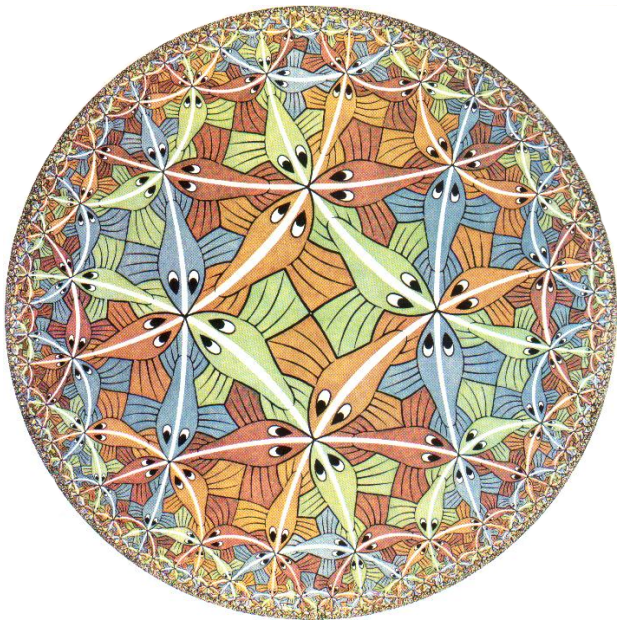

Fig. 2. Woodcut by Maurits Escher, named "Circle Limit III" (1958). After seeing Coxeter's picture [3] of the Poincaré projection of the $\mathbb{H}^2$, the artist was fascinated by the infinite space covered precisely in the disk. Note the "fish-eye" effect, seen at the fishes: all have about equal size in $\mathbb{H}^2$ – but they *appear* larger in the center.

2.3 All in one Picture: Usage of Isometric Transformations

For changing the focus point in *PD* we need a translation operation (which can be associated to a mouse click or drag event). In general, the transformations which do not change form and metric in $\mathbb{H}^2$ are, similar to $\mathbb{R}^2$ the translations, rotations, and the reflection. How do these operations look in the Poincaré *PD*? Because all $\mathbb{H}^2$-lines remain $\mathbb{H}^2$-lines, their corresponding *PD* circle arcs must remain (generalized) circle arcs in *PD*.

The transformations that respect these properties (also called "circle automorphy" transformations) are here the *Möbius* transformations $T(z)$ which can be formulated using complex numbers z, a, b: $T_{a,b}(z) = (az + \bar{b})/(bz + \bar{a})$ with $|a|^2 - |b|^2 = 1$. By describing the Poincaré disk *PD* as the complex unit circle, the isometric transformations for a point $z \in PD$ can be rewritten in the more comprehensible form:

$$z' = T(z;\ c, \theta) = \frac{\theta z + c}{\bar{c}\theta z + 1}, \quad |\theta| = 1,\ |c| < 1. \tag{3}$$

Here the complex number θ describes the pure rotation of *PD* around the origin 0. The following translation by c maps the origin to c and (without

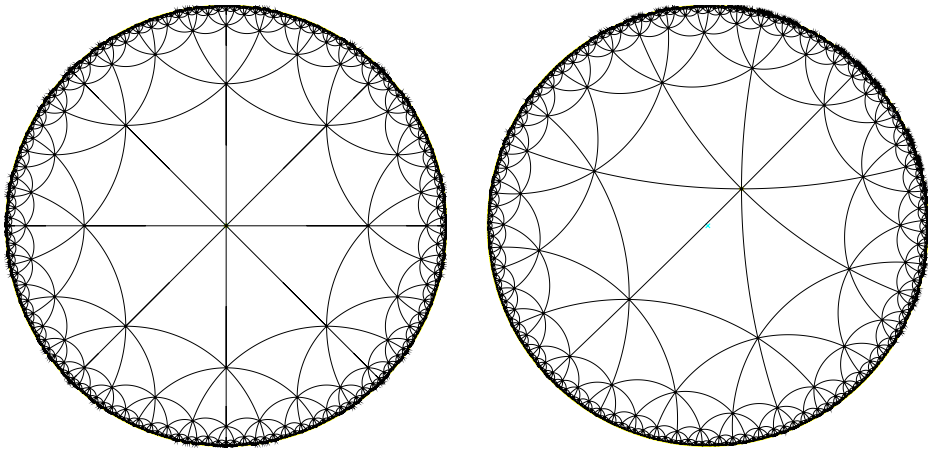

Fig. 3. *(Left:)* A perfectly symmetrical triangular $\mathbb{H}^2$ grid displayed in the Poincaré disk *PD*. While in flat R^2 a perfect triangle tessellation is build on the hexagon (6-gon) – in $\mathbb{H}^2$, tessellation with regular n-gons and $n \geq 7$ are possible. Here the 8-gon is the base structure: at each node eight equilateral triangles meet (compare Escher's woodcut Fig. 2). The edges line up straight, such that infinite long $\mathbb{H}^2$ mesh lines are constructed, which appear as (generalizes) circles in *PD*. *(Right:)* The effect of an isometric translation on the entire grid. Note, how all generalizes circles are transformed into other circles.

rotation $\theta = 1$) $-c$ becomes the new center 0. The inverse transformation is

$$T^{-1}(z;\ c, \theta) = T(z;\ -\bar{\theta}c, \bar{\theta})\ . \tag{4}$$

Two successive transformations in *PD* are computed faster by evaluating the concatenated transformation (see also [6]):

$$T(T(z;\ c_1, \theta_1);\ c_2, \theta_2) = T(z,\ c, \theta), \quad \text{with}$$
$$c = \frac{\theta_2 c_1 + c_2}{\theta_2 c_1 \bar{c}_2 + 1}, \quad \text{and} \quad \theta = \frac{\theta_1 \theta_2 + \theta_1 \bar{c}_1 c_2}{\theta_2 c_1 \bar{c}_2 + 1}$$

2.4 Does the $\mathbb{H}^2$ work like a crystal ball?

Some people explain the working principle of the $\mathbb{H}^2$ Poincaré mapping as a glass ball – a crystal ball, which can be rolled on a infinite (flat) plane. How far is this comparison meaningful?

On the positive side it can be helpful for introducing the graphical user machine interface, since many $\mathbb{H}^2$-beginners have experience with manipulating 3D scenes with a mouse. One common interaction mode is the coupling of the pitch and yaw angles of the 3D world coordinate transformation with the mouse. The concept of a turnable globe is realized and by click+drag on this

virtual ball, the visible scene can be turned and rolled, which appears very soon natural and intuitive.

This interaction concept transfers to the $\mathbb{H}^2$ since one of the most useful interaction modes is similar: clicking on a point c in PD and dragging c to another location within PD. The rest of the scene follows the non-linear Möbius transformation Eq. 3. In praxis this exposes the major advantages of the $\mathbb{H}^2$: the amount of space grows exponentially with the distance. This is apparent for the center but is a ubiquitous property. By means of the Möbius transformation this can be easily perceived. The unusual spatial growth of neighborhood includes also more angular space. As seen in Fig. 3, this allows a regular triangular tessellation of the $\mathbb{H}^2$ with more than six connecting triangles. An instructive demonstration is given in the next section.

2.5 Tree Data Accommodation and Angular Growth

Now we turn to the question raised earlier: how to accommodate data in the hyperbolic space. A solution to this question for the case of acyclic, tree-like graph data was provided by Lamping and Rao [7,6]. By using mainly successive applications of transformation Eq. 3 they developed (and patented) a method to find a suitable layout for this data type in $\mathbb{H}^2$. Each tree node receives a certain open space pie segment, where the node chooses the locations of its siblings. For all its siblings i it calls recursively the layout-routine after applying the Möbius transformation Eq. 3 in order to center i.

Fig. 4 illustrates three intermediate steps and illustrates the extraordinary behavior of angular enlargement in the $\mathbb{H}^2$. Consider the situation in the flat land: here the recursive sector partitioning divides the space in more and more acute sectors ($\approx 360°$ divided by number of leaves). Note the crucial difference to the crystal ball on the flat plane: the rolling does work, but the layout of data is limited (you would have to roll further and further).

It should be pointed out, that these ideas principally work also in higher dimensions of the hyperbolic space. E.g., Munzner developed an other graph layout algorithm for the three-dimensional hyperbolic space [9]. Her "H3Viewer" [10] allows fast drawing and frame rate interaction and can arranges trees in beautiful flower-like structures. The extra dimension opens even more space for the data layout. Unfortunately, this advantage comes with the problem of occlusions on every standard 2D display device.

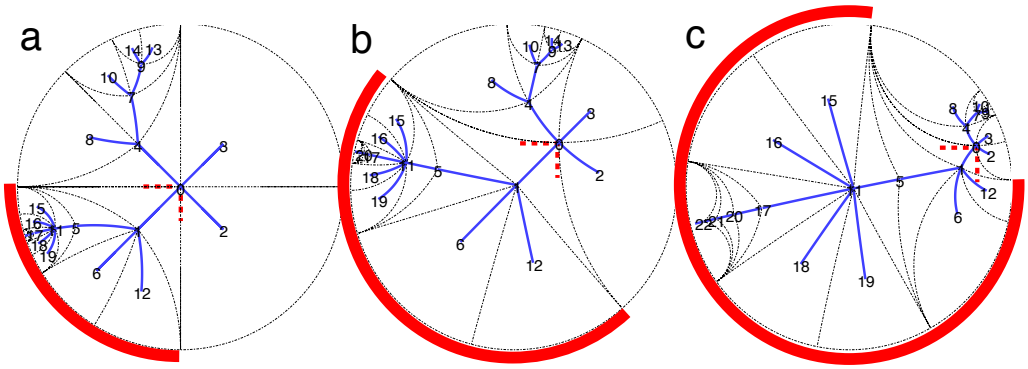

Fig. 4. *(a–c)* The $\mathbb{H}^2$ tree layout for tree-like graphs demonstrates the unusual sector area enlargement with growing neighborhood in $\mathbb{H}^2$. The dotted lines indicate the recursive construction by sector delimited by the rays (straight in $\mathbb{H}^2$, circles in *PD*). *(a)* The marked 90° pie is allocated by the root node "0" to the child node "1" (one of four). *(b)* By radial translation node "1" centers itself and distributes *its* sector among its children. Note, how the sector angle has opened up to about 180°. *(c)* Two nodes deeper, for node "11" the original 90° sector opened to more than 270°. Since the Poincaré model is conform, the angle at node "0" is preserved as indicated by the dotted angle markers.

3 Multidimensional Scaling (MDS)

First we review MDS and the Sammon algorithm.

Multidimensional scaling refers to a class of algorithms for finding a suitable representation of *proximity* relations of N objects by distances between points in a low dimensional – usually Euclidean – space. For a detailed analysis on proximity structures, see [1]. In the following we represent proximity as *dissimilarity* values between pairs of objects, mathematically written as dissimilarity $\delta_{ij} \in \mathbb{R}_0^+$ between the i and j item. As usual we assume symmetry, i.e. $\delta_{ij} = \delta_{ji}$. Often the raw dissimilarity distribution is not suitable for the low-dimensional embedding and an additional δ-processing step is applied. We model it here as a monotonic transformation D(.) of dissimilarities δ_{ij} into *disparities* $\mathrm{D_{ij}} = \mathrm{D}(\delta_{\mathrm{ij}})$. In Sec. 6.3 we will discuss a transformation of adjusting a potential dimensionality mismatch.

The goal of the MDS algorithm is to find the spatial representation $\mathbf{x}_i$ of each objects i in the L-dimensional space, where the pair distances $d_{ij} \equiv d(\mathbf{x}_i, \mathbf{x}_j)$ match the disparities $\mathrm{D_{ij}}$ as faithfully as possible $\forall_{i \neq j} \mathrm{D_{ij}} \approx \mathrm{d_{ij}}$. Here, L is typically two or three, since again the main purpose of MDS is visualization and explorative data analysis. The pair distance is usually measured by the Euclidian distance:

$$d_{ij} = ||\mathbf{x}_i - \mathbf{x}_j|| \qquad \text{with} \quad \mathbf{x}_i \in \mathbb{R}^L, \; i,j \in \{1,2,..N\} \tag{5}$$

3.1 Sammon's Algorithm

One of the most widely known MDS algorithms was introduced by Sammon [17] in 1969. He formulates a minimization problem of a cost function which sums over the squares of disparities–distance misfits, here written as

$$E(\{\mathbf{x}_i\}) = \sum_{i=1}^{N} \sum_{j>i} w_{ij} (d_{ij} - \mathrm{D_{ij}})^2. \tag{6}$$

The factors w_{ij} are introduced to weight the disparities individually and also to normalize the cost function E to be independent to the absolute scale of the disparities $\mathrm{D_{ij}}$. Depending on the given analysis task the factors can be chosen to weight all the disparities equally – the *global* variant ($w_{ij}^{(g)} = const$) – or to emphasize the *local* structure by reducing the influence of larger disparities ($w_{ij}^{(l)}$)

$$w_{ij}^{(g)} = \frac{1}{\sum_{k=1}^{N} \sum_{l>k} \mathrm{D_{kl}^2}} \;, \qquad w_{ij}^{(l)} = \frac{2}{N(N-1)} \frac{1}{\mathrm{D_{ij}^2}} \;. \tag{7}$$

Note that the latter is undefined if any pair has zero disparity. In his original work [17] Sammon suggested an *intermediate* normalization

$$w_{ij}^{(m)} = \frac{1}{\sum_{k=1}^{N} \sum_{l>k} \mathrm{D_{kl}}} \frac{1}{\mathrm{D_{ij}}} \tag{8}$$

which we are using in the following. Sammon proposed a steepest gradient method, in particular, the (diagonal) Newton method to iteratively minimize the remaining cost or stress E. He ignored the off-diagonal part of the Hessian matrix and used a step length reduced by a "magic" factor η of 0.3–0.4. Starting from a random $\{\mathbf{x}_i\}$ initialization, in each iteration step, one object i^* is considered, and Eq. 6 minimized with respect to $\mathbf{x}_{i^*}$

$$\mathbf{x}_{i^*}^{(new)} = \mathbf{x}_{i^*}^{(old)} + \eta \Delta_{i^*} \tag{9}$$

with Δ_{i^*} here written per component $q \in \{1, .., L\}$

$$\Delta_{i^*,q} = -\frac{\partial E}{\partial x_{i^*,q}} \Big/ \left| \frac{\partial^2 E}{\partial x_{i^*,q}^2} \right| . \tag{10}$$

The algorithm usually needs several epochs (with random sequence of selected i^*) to converge to an cost-function minimum. The usual methods to deal with

the risk of converging to a local minimum are restarting with different initial conditions and selecting the result with the lowest cost function or stress E. To save CPU time one may want to start with a good initial configuration $\{\mathbf{x}_i\}$ for example by using the first L principal components found by PCA.

The reader is referred to [2] for further details on this and other MDS algorithms.

4 Multidimensional Scaling in $\mathbb{H}^2$

How can the data be accommodated in the hyperbolic plane – preserving the distance structure of the given data? We seek the synthesis of the MDS-approach and the visualization capabilities of the hyperbolic plane. The core idea turn out to be very simple: instead of finding a MDS solution in the low-dimensional Euclidean $\mathbb{R}^L$ and transferring it to the $\mathbb{H}^2$ (which can not work well), the MDS formalism operates in the hyperbolic space from the beginning. The key point is Eq. 5. The Euclidean distance in the target space is replaced by the appropriate distance metric for the Poincaré model (see, e.g. [8])

$$d_{ij} = 2 \operatorname{arctanh}\left(\frac{|\mathbf{x}_i - \mathbf{x}_j|}{|1 - \mathbf{x}_i \bar{\mathbf{x}}_j|}\right) \quad , \quad \mathbf{x}_i, \mathbf{x}_j \in PD. \tag{11}$$

Several aspects need consideration:

The PD disk is the entire $\mathbb{H}^2$. Therefore, care must be taken to stay within the unit circle when updating the point i^*. But a simple clipping or shrinkage of the update step $\eta\Delta_{i^*}$ in Eq. 9 is not appropriate. Instead the Möbius transformation Eq. 3 leads in the right direction and handles the rapid shrinkage in the vicinity of the circle rim.

$$\mathbf{x}_{i^*}^{(new)} = T(\mathbf{x}_{i^*}^{(old)};\ \eta\Delta_{i^*}, 1); \tag{12}$$

While the gradients $\partial d_{ij,q}/\partial x_{i,q}$ required in Eq. 10, are rather simple to compute for the Euclidean case ($= (x_{i,q} - x_{j,q})/d_{ij}$), the case becomes complex for Eq. 11:

$$\frac{\partial}{\partial x_{i,1}} d(\mathbf{x}_i, \mathbf{x}_j) = \frac{2t}{1-t^2}\left(\frac{v_1}{v_1^2 + v_2^2} - \frac{x_{j,1}v_3 + x_{j,2}v_4}{v_3^2 + v_4^2}\right) \tag{13}$$

$$\frac{\partial}{\partial x_{i,2}} d(\mathbf{x}_i, \mathbf{x}_j) = \frac{2t}{1-t^2}\left(\frac{v_2}{v_1^2 + v_2^2} + \frac{x_{i,1}v_4 - x_{j,2}v_3}{v_3^2 + v_4^2}\right) \tag{14}$$

with

$$
\begin{aligned}
\mathbf{x}_i &= (x_{i,1} + \mathrm{i}\, x_{i,2}) \ \in PD \subset \mathbb{C} \\
\mathbf{x}_j &= (x_{j,1} + \mathrm{i}\, x_{j,2}) \ \in PD \subset \mathbb{C} \\
v_1 &= x_{i,1} - x_{j,1} \\
v_3 &= x_{i,1}x_{j,1} + x_{i,2}x_{j,2} - 1 \\
v_2 &= x_{i,2} - x_{j,2} \\
v_4 &= x_{i,1}x_{j,2} - x_{j,1}x_{i,2} \\
t^2 &= \frac{v_1^2 + v_2^2}{v_3^2 + v_4^2}
\end{aligned} \tag{15}
$$

This bears two critical cases:

- the denominator $v_1^2 + v_2^2$ becomes zero only if the points $\mathbf{x}_i$ and $\mathbf{x}_j$ are equal;
- $v_3^2 + v_4^2 = 0$ does not occur for valid points inside the unit circle.

Due to the complexity of these results we choose to drop the computation of the second derivatives and rather improve the cost minimization step Eq. 10. We employed the Levenberg-Marquardt approach and followed the argumentation in [13] by dropping the higher derivatives. The implementation uses for each data point i an individual and bounded parameter $\lambda \in [10^{-4}, 10^4]$.

4.1 Disparity preprocessing

Due to the non-linearity of Eq. 11 the preprocessing function D(.) (see Sec. 3.1) has more influence in $I\!H^2$. Consider, e.g., linear rescaling of the dissimilarities

$$
\mathrm{D}_{\mathrm{ij}} = \mathrm{D}(\delta_{\mathrm{ij}}) = \alpha\, \delta_{\mathrm{ij}} \qquad \alpha > 0. \tag{16}
$$

In the Euclidean case the visual structure is not affected – only magnified by the factor α. In contrast in $I\!H^2$, α scales the amount of curvature that is felt by the data. In the *PD* this has the effect that the increasing α shifts all points further outside – but there is also much more (exponentially more) space. The optimal α depends on the given dataset, its dissimilarity structure, and last – not least – on the given visualization task.

In the experiments reported below, we report on the effect of tuning α. For high-dimensional datasets we discuss in Sec. 6.3 also nonlinear transformations in order to enhance the disparity contrast in the visual display of a dataset.

5 Introductory Experiments

We conducted experiments and present several real and synthetic datasets.

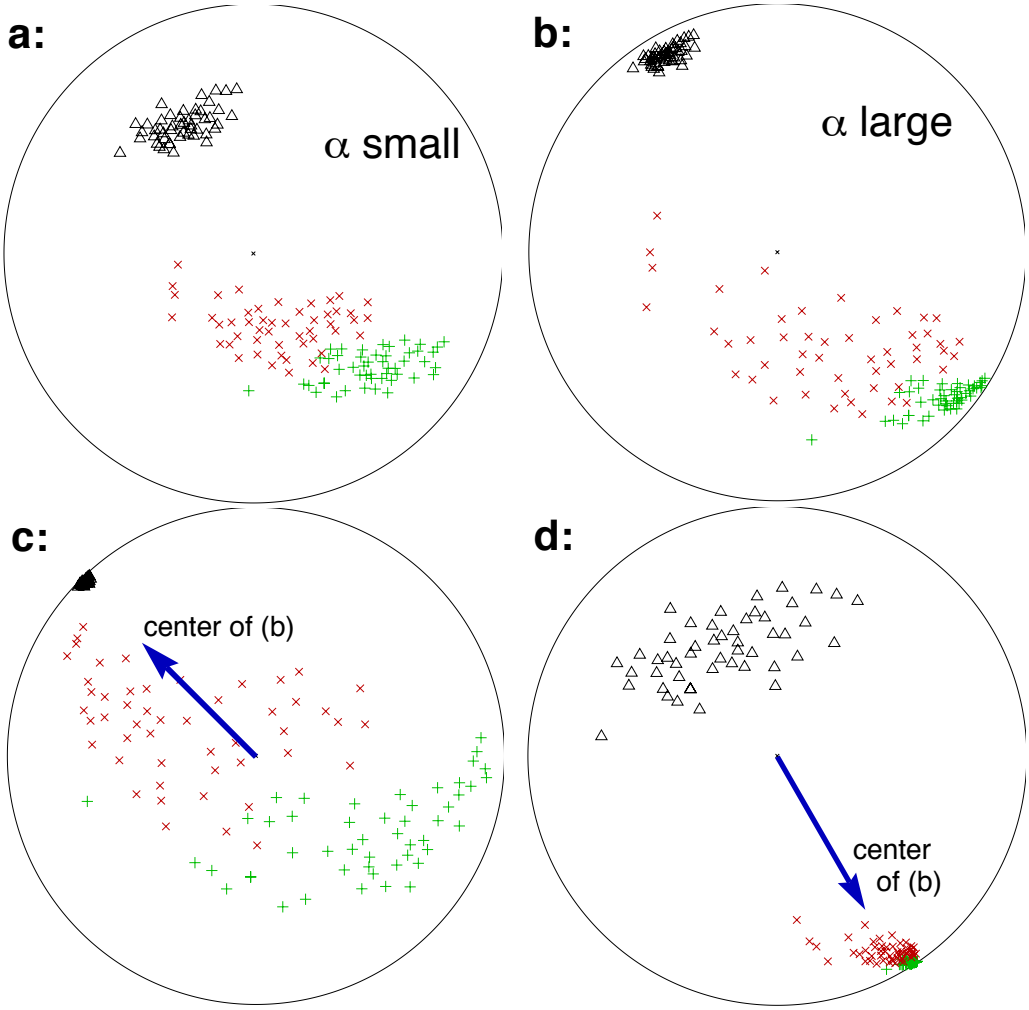

Fig. 5. *(a-d) Iris dataset*: the three iris classes are clustered and well separated in *PD* as indicated by their markers. *(a)* uses a smaller α for dissimilarity rescaling than *(b)*. While *(a)* looks comparable to a common Sammon map in $\mathbb{R}^2$, in *(b–d)* the wider, exponentially growing space in $\mathbb{H}^2$ is used and can be explored by moving the focus point (via mouse drag) to the areas of interest. The arrows indicate the original focus location in *(b)*.

5.1 Iris Dataset

Fig. 5(a-d) displays the obligatory "Hello World" example of data mining – Fisher's *Iris* data set. It describes 150 flower samples of three types: *iris setosa* ("Δ"), *iris versicolour* ("$\times$"), and *iris virginica* ("+"). δ is here the Euclidean pair distance in the four describing components sepal length, sepal width, petal length, and petal width. The clear separation of the class demonstrates that MDS performs also well in the hyperbolic plane. Several navigation snapshots visualize the focus + context effect with three navigation snapshots for two different α values.

5.2 Animals Dataset

Fig. 6 presents the second dataset and demonstrates the preservation of semantic proximity. The dissimilarity measure is the euclidean distance in the 13 features (see caption and UCI Machine Learning Repository). Herewith, the more properties are shared by a pair of animals, the closer they should settle. The semantic relationship appears very clearly in the H-MDS display.

Fig. 7 displays the rest stress $E(\alpha)$ depending on the scaling factor α in Eq. 16.

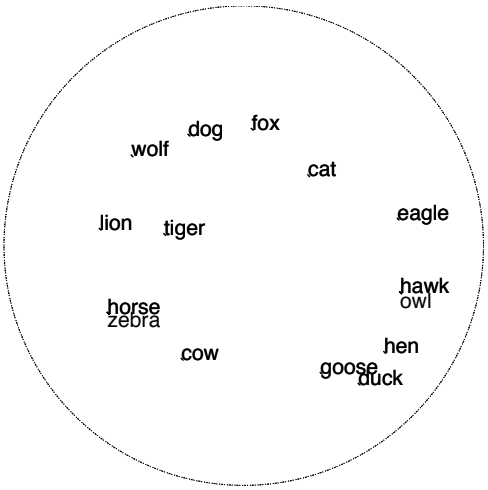

Fig. 6. The **animals dataset** describes 16 animals by 13 biological, binary coded features (small, medium, big, 2 legs, 4 legs, hair, hooves, mane, feathers, hunter, runner, can fly, and swimmer). Two pairs have equal features and are mapped on the same spots: *owl+hawk* at 4 o'clock and *horse+zebra* at 8 o'clock.

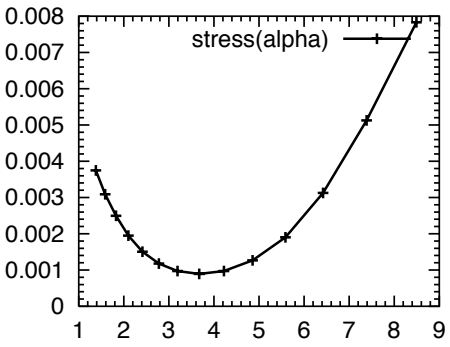

Fig. 7. The remaining stress $E_{I\!H^2}$ for the animals dataset versus the dissimilarity scaling α: The minimum of $E_{I\!H^2} = 0.00089$ at $\alpha = 3.7$ is less than a fifth of the rest stress for the flat embedding $E_{I\!R^2} = 0.0048$.

5.3 Random Tree in 200 Dimensions

Fig. 8 shows a hierarchical cluster dataset with 280 points. It comprises the nodes of a three-level "random tree" reaching out in the $I\!R^{200}$. The directions of the branches were chosen randomly, while the branching factors were 5-5-10 and the link lengths were 2-1.5-0.5.

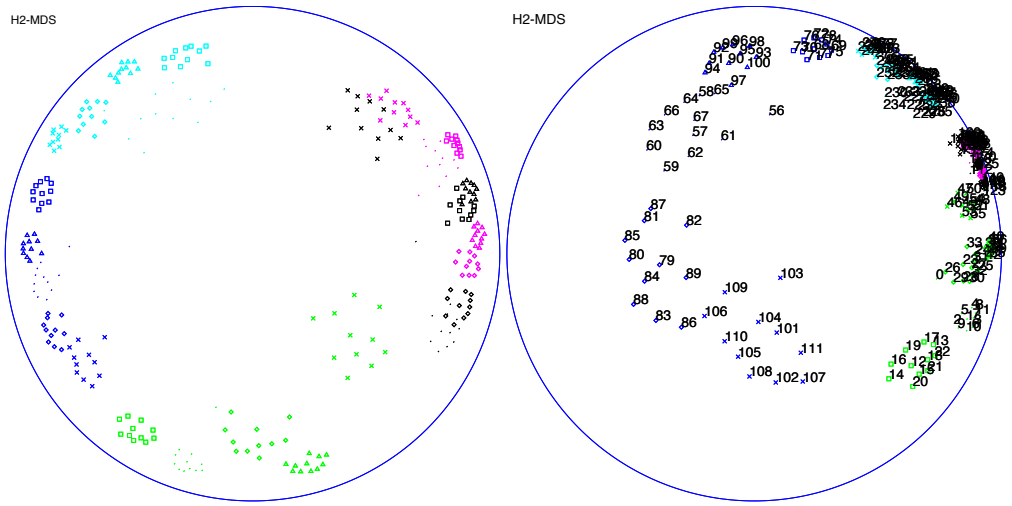

Fig. 8. The three-level "random tree" in the $\mathbb{R}^{200}$ MDS projected in the $\mathbb{H}^2$. Displayed are snapshots of navigation through the mapping, with labeling turned on and off (systematically labeled by three digits according to the branch numbers).

For identification the nodes can be optionally labeled. By using the $\mathbb{H}^2$ focus+context navigation one can interactively inspect the details where the labels were overlapping in the initial focus position. Fig. 8 display two snapshots of this process. However, this can only give a glimpse of the dynamic look and feel of the life interaction.

How advantageous is the hyperbolic embedding compared to the euclidean embedding? To answer this question we examine the stress $E(\{\mathbf{x}_i\})$ (Eq. 6) depending on the scaling factor α (Eq. 16). With growing α the data experiences more and more negative curved space. In Fig. 9 (*top-left*) we find rest stress E has a minimum $E_{\mathbb{H}^2\text{min}} = 0.0095$ at $\alpha = 2.9$ which compares to a classical Sammon mapping in $\mathbb{R}^2$ with a more than five times higher rest stress $E_{\mathbb{R}^2\text{min}} = 0.057$.

For the optimal α-point, Fig. 9 (*right*) displays also the relation between the resulting pair distances versus the scaled disparities as scatterplot (D_{ij}, d_{ij}). A least square regression line is fitted and returns a Pearson's correlation coefficient of $R_{H^2} = 0.901$. This compares favorably to $R_{R^2} = 0.702$ for the Sammon algorithm in R^2. The *bottom-left* curve displays the developement of the correlation coefficient $R(\alpha)$ with growing dissimilarity scaling α. This demonstrates that the dataset can profit from the non-euclidean geometric embedding.

How does this compare to conventional visualizations? Fig. 8a shows the same dataset *(left)* as random projection and Fig. 8b as principal axis projection. While the cluster structure appears clearly in the $\mathbb{H}^2$ mapping Fig. 8, this is not the case, neither for a random projection, nor for the PCA-projection in direction of the two strongest eigenvectors Fig. 10.

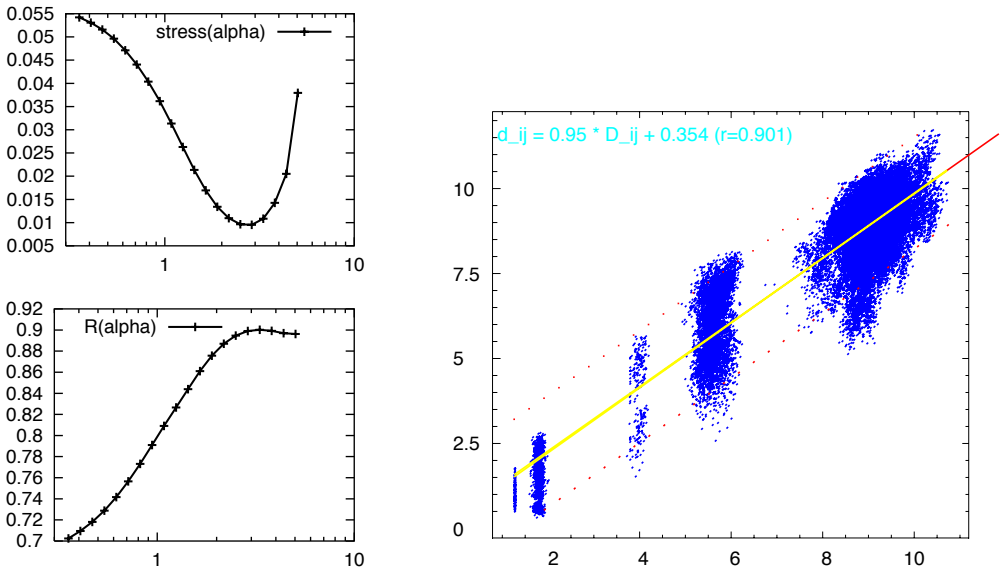

Fig. 9. *(a, top-left)* The rest stress $E_{I\!H^2}$ versus the dissimilarity scaling α shows a minimum at $\alpha = 2.9$. *(b right:)* Scatterplot of the pair distances d_{ij} in $I\!H^2$ versus the α scaled dissimilarities D_{ij}. The plot has a good linear regression fit with the correlation coefficient $R = 0.901$ for $\alpha = 2.9$ *(c bottom-left)* The correlation coefficient $R(\alpha)$ has an $E(\alpha)$-corresponding maximum.

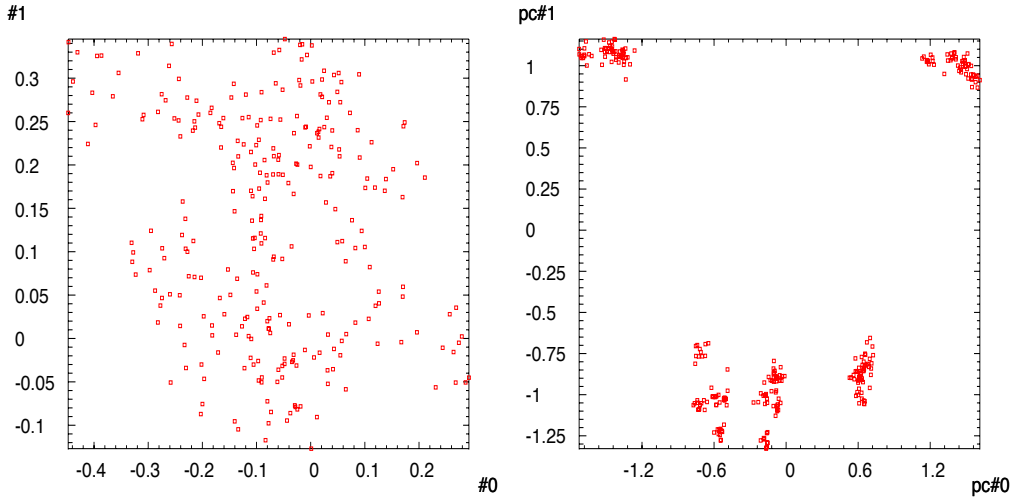

Fig. 10. *(a left:)* Random projection of the "random tree" in the $I\!R^{200}$. *(b right:)* PCA-projection to plane spanned by the two largest eigenvectors. The precise cluster structure remains hidden. Note, a prerequisite of both displays is that the data is available as vector in $I\!R^M$ – this is not the case for the MDS.

5.4 Random Distributions in High-Dimensions

Feature-rich datasets are inherently high-dimensional. With growing dimensionality most of their volume is contained in a thin, outer shell. This well

know characteristics has consequences for the proximities distribution of high-dimensional datasets in general. Using Monte-Carlo simulations we examined several random M-dimensional point distributions with *(i)* M i.i.d. Gauss random variables for each component, *(ii)* uniform distribution inside the unit sphere, *(iii)* on its surface, *(iv)* inside the M-dimensional unit hypercube and *(v)* on its corners.

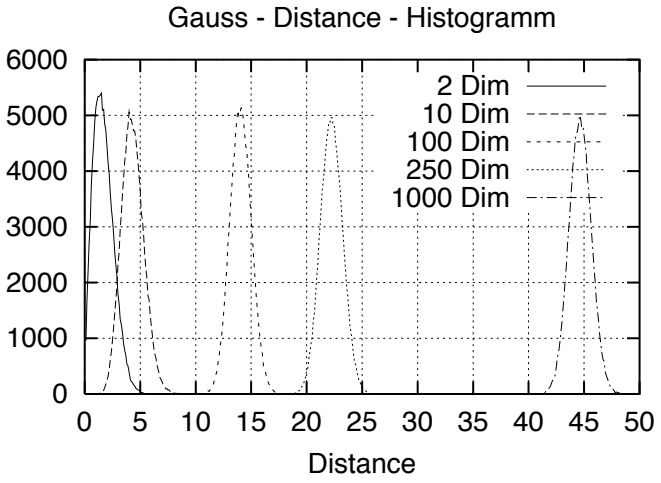

Fig. 11. Histogram of euclidean distances δ between pairs from a radial Gauss distribution in various dimensions $M \in \{2, 10, 100, 250, 1000\}$ (124,750 pairs from 500 random points with unit variance; binning width is 0.1).

Fig. 5.4 illustrates the first case as histogram of the pairwise distances. With growing M, the distance distribution δ shifts to larger distances without significant widening. For the other cases *(ii-v)* the principal structure is the same except for some peak structures details for low dimensions (small M). This suggest, the observation holds also for other cases and can be generalized to inherently high-dimensional data, i.e. not laying in a low-dimensional sub-space.

Consequences are, as noted by several other authors [4,5], that the optimal embedding in low-L spaces tends to become a circular ring structure for growing M-dimensionality. Unfortunately, the situation bears numerous local minimal and the plain Sammon algorithm often returns sub-optimal results (some available implementations exhibit serious numerical problems and return here bizarre structures, e.g. the *sompak*-package).

How does the $\mathbb{H}^2$ accommodate such a dataset? Fig. 12 shows the two target mappings for *(left)* $\mathbb{R}^2$ and *(right)* $\mathbb{H}^2$ together with their disparity–distance scatterplots $\{\mathrm{D_{ij}}, \mathrm{d_{ij}}\}$ ($\alpha = 0.33$). The remaining stress was 40% higher for the Euclidean embedding with $E_{\mathbb{R}^2} = 0.40$ versus $E_{\mathbb{H}^2} = 0.285$, which is comprehensible also in the two (d_{ij}, D_{ij}) scatterplots in Fig. 12(c,d).

It turns out, that the $\mathbb{H}^2$ is not free of local minimal – but circumstances are much better than in $\mathbb{R}$: obviously the cost function landscape offers more space to circumvent local minima. And solutions can be found more easily in the hyperbolic geometry.

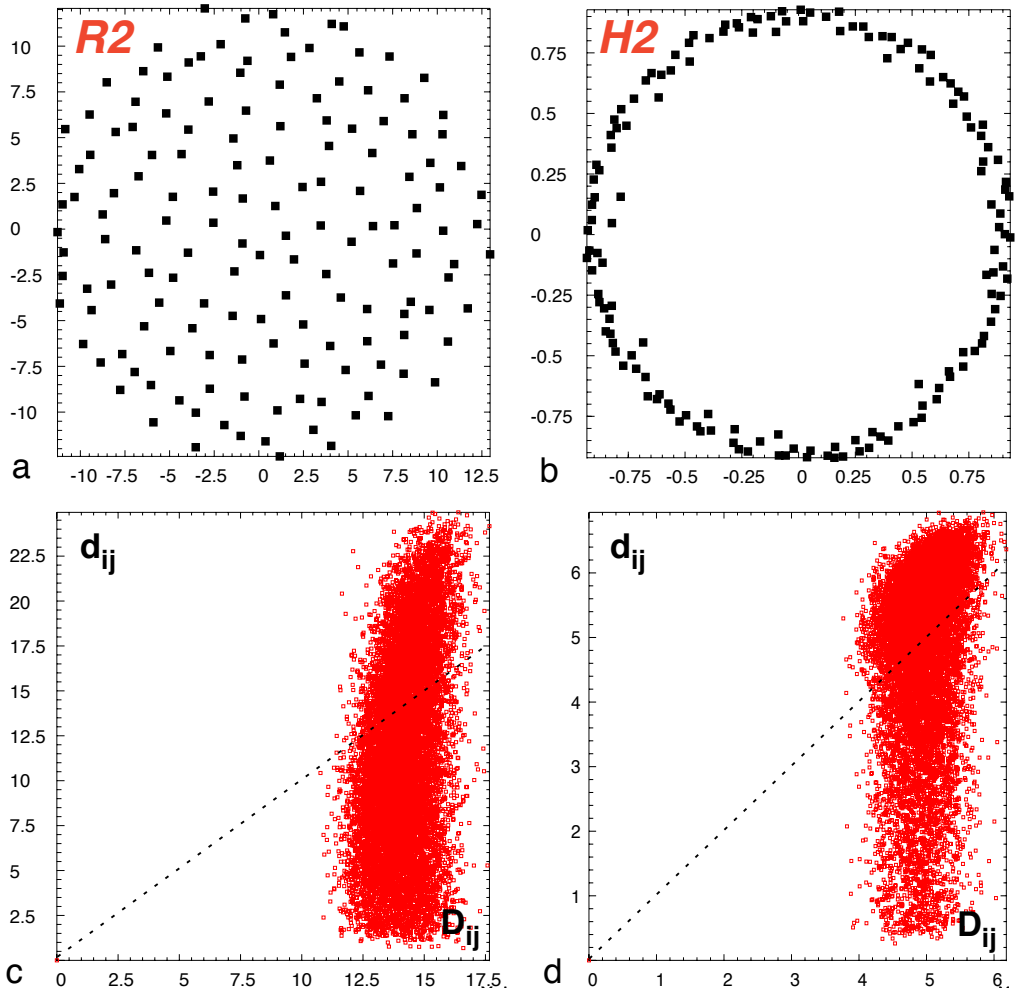

Fig. 12. *(Top)* Multi-dimensional scaling of 150 Gauss distributed points in $M = 100$ dimensions into the *(a)* $\mathbb{R}^2$ and *(b)* into the hyperbolic $\mathbb{H}^2$. *(Top:)* The MDS in Euclidian space *(a)* gets more easy locked in local minima than the MDS in the hyperbolic space. The right side *(b)* shows the ring-like structure previously reported to be more optimal for this high-dimensional mapping task. Probably the extra space in $\mathbb{H}^2$ allows to circumvent around the local minima during the iterative MDS process. *(Bottom:)* The two corresponding scatterplot exhibit the target dissimilarities D_{ij} and the obtained distances d_{ij}. The optimal distribution has little rest stress and approximates the dashed diagonal line. *(d)* In $\mathbb{H}^2$ the majority of points lay closer than in the left diagram *(c)* which tends to smaller d_{ij} below the diagonal.

Later in Sec. 6.3, we will return to the display of high-dimensional data as "ring" structures and measurements to modify its appearance.

6 Application to Navigation

6.1 Bag of Words – Standard Representation for Text Mining

In the domain of the information retrieval and text mining text is very often treated as *bag-of-words* and represented as very high-dimensional vector. One may argue that this ignores completely semantic information word order – amazingly the results justify this drastic step.

Given a collection of N text documents, first a vocabulary – a set of words $\{w_i\}$ is determined. Words in text are always preprocessed by a suitable *word stemming* procedure and filtering out of *stop words*. The vocabulary is then the interesting part of all unique word stems, i.e. the most and the least frequent words are rejected. Each text document t is represented by a feature vector $\vec{f}_t$, where the components $f_{t,i}$ are determined by

$$f_{t,i} = TF(t, w_i) \log\left(\frac{N}{DF(w_i)}\right). \tag{17}$$

$TF(t, w_i)$ is the *term frequency* and counts the number of times the term w_i occurred in document t. $DF(w_i)$ denoted the *document frequency* and counts the number of documents where the term occurred. This standard weighting scheme emphasizes rare words as more significant than common words, for more details see [16]. Proximity $(= 1 - \delta)$ and therewith dissimilarities of two documents is computed with the cosine metric

$$\delta_{ij} = 1 - \cos(\vec{f}_{t_i}, \vec{f}_{t_j}) = 1 - \vec{f}'_{t_i}\vec{f}'_{t_j}, \text{ with } \vec{f}' = \frac{\vec{f}}{||\vec{f}||} \tag{18}$$

and efficiently implemented by storing the normalized document feature vectors $\vec{f}'$.

6.2 Computing the Movie Representation:

As an example we used a text collection consisting of film critiques taken from the rec.art.movies.reviews newsgroup. For the purpose of identification extra information, such as the film genre classification, is drawn from the internet movie database (http://www.imdb.org). The vocabulary with 5084 distinct terms was extracted and the feature vectors $\vec{f}_t$ computed (for more details consult [11]). Based on this intermediate review database we selected

all movies of the categories *animation* and *science-fiction* which had more than five recorded reviews. Then, each *average movie feature vector* was calculated from the set of its associated reviews

$$\vec{f}_m = \sum_{t \in A} \vec{f}_t \,, \quad A = \{t \,|\, t \;\text{ describing movie } m\}. \tag{19}$$

In contrast to the usage of directly accessible, brief movie plot description this procedure increases not only the amount of source information but also increases author independence by averaging.

6.3 Two Examples of Proximity Contrast Modulations

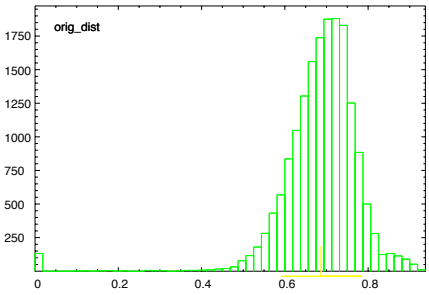

Fig. 13. *(top:)* Histogram of dissimilarities δ_{ij} for the selected movies.

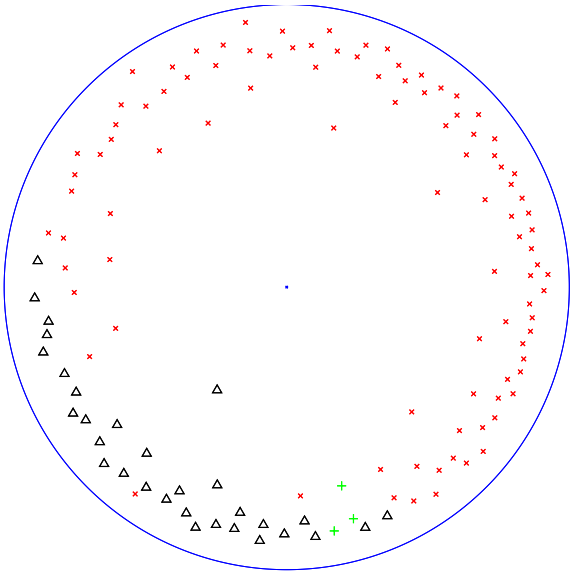

Fig. 14. *(right:)* Hyperbolic MDS result for the linear disparity transformation Eq. 18 and Eq. 16. The typical "ring" structure is visible due the high dimensional (cmp. Fig. 12). The red "×" mark *science-fiction*, the black "Δ" *animation*, and the green "+" mark films belonging to both genres. Note, this genre information was used in post processing but not included in the information input.

How does the dissimilarity structure of the 132 selected, well reviewed movies compare to a high-dimensional Gauss-distribution? Fig. 13 shows the histogram of the obtained 8646 dissimilarities. Comparing them to Fig. 5.4 supports that the "effective" dimensionality is lower than the 5084 dimensions of the movie feature vectors $\vec{f}$. Fig. 14, displays the H-MDS picture with the typical ring structure previously discussed. The markers allow a visual assessment of the mapping quality: While the genre information was not included in the input, the H-MDS approach found a clear genre separation from the text corpus: the "Δ"-marked animation films are well separated from the

"×"-marked science-fiction group. The three movies, "+"-marked belong to both genre and became indeed located in the border zone.

While the "ring" structure is over all the most distance preserving – for visualization purpose, it is not necessarily the most advantageous to navigate in a ring of object, or here, a ring of movies.

As seen before in Fig. 5.4, the higher the dimensionality, the more the pair distance distribution is systematically shifted to larger values, whcih causes the ring appearance. Inspired by work from [5] we experimented with counter measures. The basic idea is a non-linear transformation D(.), which shifts effectively the high-dimensional dissimilarity modes back to lower effective dimensions. This enlarges the relative differences of pair distances and can therefore be considered as a *contrast enhancement*. In the following two functional concept are examined.

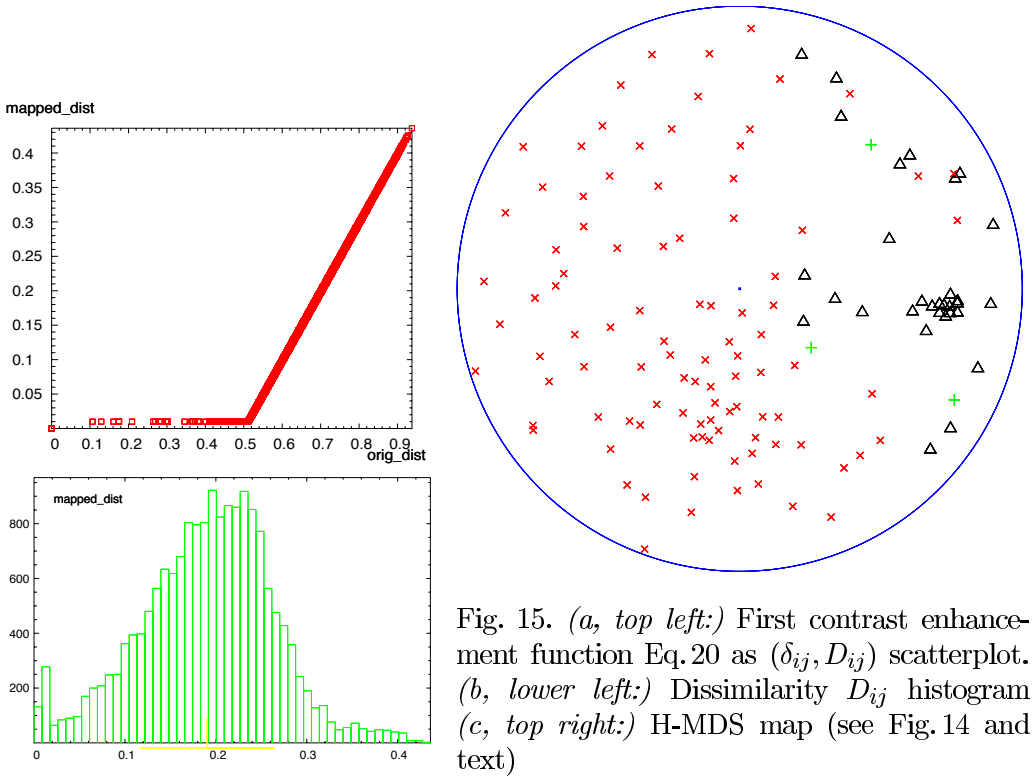

Fig. 15. *(a, top left:)* First contrast enhancement function Eq. 20 as (δ_{ij}, D_{ij}) scatterplot. *(b, lower left:)* Dissimilarity D_{ij} histogram *(c, top right:)* H-MDS map (see Fig. 14 and text)

The first is a linear dissimilarity shift with clipping to a minimal value $D_A > 0$

$$\mathrm{D_{ij}} = \alpha \ \max(\ \delta_{\mathrm{ij}} - \delta_{\mathrm{A}}\ ,\ \mathrm{D_A}). \tag{20}$$

Fig. 15a displays the transfer function for $\delta_A = Q_{\delta^+}(0.01)$ equal the 1 %percentile of the δ^+_{ij}-distribution (ignoring all $\delta_{ii} = 0$ elements). The clipping to a positive minimum $D_A = 0.01$ provides a minimal repulsive force between data pairs. In the resulting dissimilarity distribution D_{ij} (see histogram Fig. 15b) a

clear peak is visible (containing 1 % of non-zero pairs). The H-MDS result in Fig. 15c lost the ring hole (Fig. 14 and exhibits some narrow clusters.

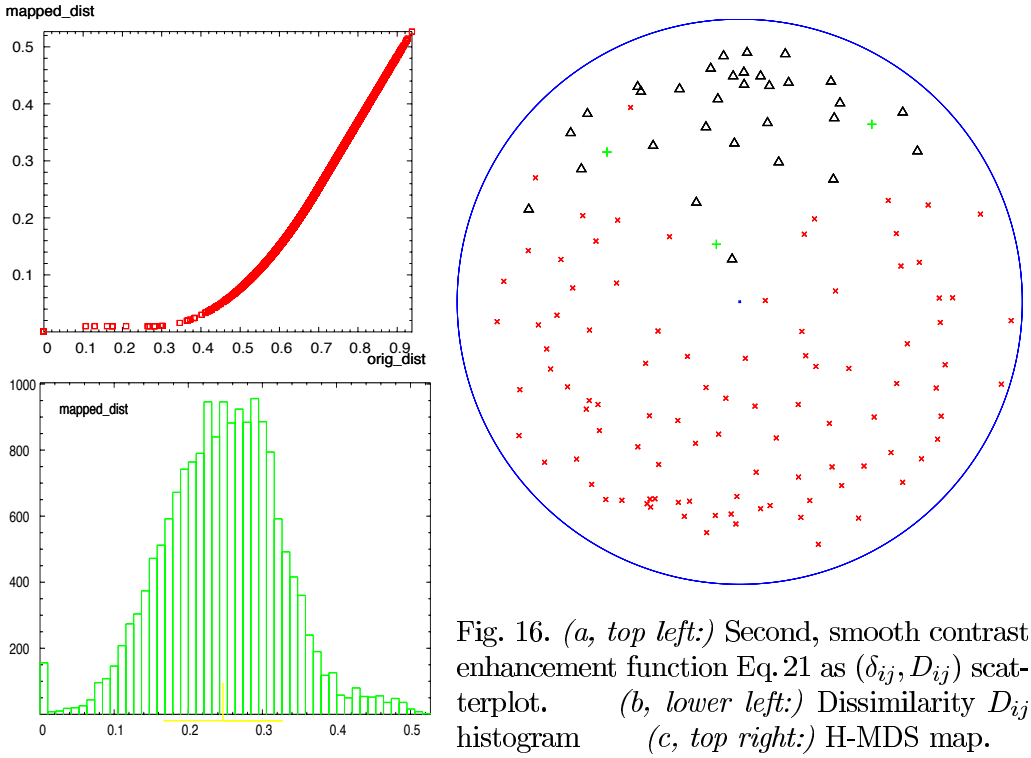

Fig. 16. *(a, top left:)* Second, smooth contrast enhancement function Eq. 21 as (δ_{ij}, D_{ij}) scatterplot. *(b, lower left:)* Dissimilarity D_{ij} histogram *(c, top right:)* H-MDS map.

The second transformation aims at a smooth, edge-free transition. The function is a piecewise polynomial

$$D_{ij} = \alpha \begin{cases} D_A & \text{if } 0 < \delta_{ij} \le \delta_A \\ D_A + c_1(\delta_{ij} - \delta_A)^2 & \text{if } \delta_A < \delta_{ij} \le \delta_B \\ c_2 + c_3\,\delta_{ij} & \text{if } \delta_B < \delta_{ij} \end{cases} \tag{21}$$

which shows smooth transitions at the connecting control points $A = (\delta_A, D_A)$ and $B = (\delta_B, D_B)$ with

$$c_1 = \frac{D_B - D_A}{(\delta_B - \delta_A)^2}, \quad c_2 = D_B - \delta_B c_3, \quad \text{and} \quad c_3 = 2\frac{D_B - D_A}{\delta_B - \delta_A}. \tag{22}$$

Fig. 16a displays the two linear pieces and the middle square connection for $A = (Q_{\delta^+}(0.001), 0.01)$ and $B = (Q_{\delta^+}(0.5), 0.25)$. The resulting dissimilarity distribution in Fig. 16b is smoother also in the low D region. The H-MDS Fig. 16c mapping is a reasonable compromise between roughly uniform distribution with good visibility and navigability of each object. Compared to Fig. 15 the smooth transition version displays a wider cohesion and improved

purity of the "Δ"-marked animation genre. The narrow cluster, mostly Disney World cartoons (see also the fully labeled snapshots in Figs. 18ab, 19ab) become wider and better distinguishable. But still, clear micro clusters are visible, giving insight to extraordinary film relations. E.g., the top left Δ pair (11 o'clock) represent "Toy Story 1+2", the ×-cluster at 7 o'clock can be identified as four "Alien" films. See Figs. 18ab, 19ab for further snapshots with labeling turned on.

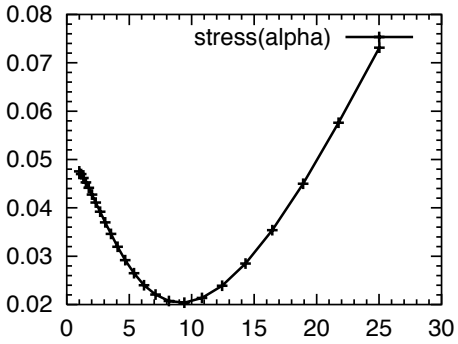

Fig. 17. The rest stress $E_{I\!H^2}$ versus the dissimilarity scaling α in Eq. 21 shows a minimum at $\alpha = 9.4$. This shows the natural advantage of the MDS embedding in the hyperbolic space with its properties of an locally, exponentially growing neighborhood.

Finally, Fig. 17 answers the question whether an hyperbolic embedding of this data set is advantageous. Yes, it is, as seen at the stress $E(\alpha)$ minimum for $\alpha = 9.4$. If the dissimilarity structure would prefer a Euclidean flat embedding then the stress minimum would appear at a diminishing small α level.

7 Discussion and Conclusion

The snapshots in Figs. 18,19 give a glimpse at the potential exploratory usage of the introduced H-MDS, the hyperbolic multi-dimensional scaling technique.

Obvious clusters like the "Star Trek" group, or the proximity of "Bug's Life" and "Antz" strike. Cineastes may want to mine further examples in the appendix figures. Note, that no extra information was imposed. The spatial structuring was automatically determined, driven by the (dis-)similarities of word frequencies in secondary, descriptive texts of totally unrelated authors.

These results show that the H-MDS approach is well suited to support "semantic browsing" in datasets or document collections. Compared to the HSOM approach with a fixed grid [11], the H-MDS freely builds a spatial structure and can therefore display the closeness of objects – as seen, e.g., in the tight "Alien" cluster.

Through the choice of the dissimilarity transformation $D(\delta)$ *(i)* the amount and resolution of the visible information can be adjusted by scaling α in Eq. 16; This extra degree-of-freedom can be used to choose a compromise between

Fig. 18. Four navigation snapshots (cont. in next Fig. 19) through the developed "space of movies". *(a top:)* The mapping is similar to Fig. 16 but with labeling turned on. *(b bottom:)* By shifting the focus to the upper *science-fiction* region ('× marked) one can discover, e.g., a band of "Star Trek" movies, around 11 o'clock. Now the tight "Alien" cluster can be inspected: at 2 o'clock four movies of this series gathered very close.

H2-Sammon

Fig. 19. (cont. Fig. 18) two further navigation snapshots in the developed "space of movies". *(a top:)* The focus got dragged into the lower left corner – in the area where the *animation* movies concentrated (marked by 'Δ). *(b bottom:)* A further closeup to a larger group of related Disney animation movies can be identified.

visibility of the entire structure and space for navigation in the detail-rich outer areas. It is certainly possible to integrate the optimization of α in the overall minimization. However, for mapping problems with $\mathbb{R}^2$ topology, this would obviously lead to zeroing α, which is not intended.

(ii) Non-linear dissimilarities transformations can serve as contrast enhancement. The piecewise composed Eq. 20 or Eq. 21 illustrate how the inherent distance properties of high-dimensional distribution can by compensated. They shift the elevated distances to lower values and regulate the base repulsion between data pairs. The result is a wider spread-out distribution in $\mathbb{H}^2$ without ring hole and tighter clusters. The compactness of the clusters can be accommodated to the desired detail visibility and readability of the mapping result.

From MDS the H-MDS inherits the efficiency scaling behavior of $N(N-1)/2$ required dissimilarities for N points. On the positive side it also allows to process non-vectorial data, available only as pairwise proximities.

From the embedding in $\mathbb{H}^2$ this new approach profits not only from the extra space for compressing semantic relationships. It also gains the superb visualization and navigation properties, which were found to yield significant improvement in user task time compared to traditional browsing methods.

Furthermore the embedding of the data in the hyperbolic space reduces the overall stress compared to a conventional, flat embedding in the $\mathbb{R}^2$ as seen, e.g., in Figs. 7, 9, and 17. This advantage is certainly a consequence of the exponential growth of the neighborhood around each point. It suggests the hyperbolic plane as a good target space for a natural embedding of high-dimensional data. An interesting side observation is, that the hyperbolic space also seems to be a better terrain for circumventing local minima in a gradient descent process.

Future work will address *(i)* the automatic customization of the proximity transformation, *(ii)* the combination of the H-MDS with the HSOM for visualizing very large datasets. *(iii)* When looking at the label intense appendix figures, it appears obvious that the display can be successfully combined with various search operations and suitable graphical attribution.

References

[1] R.M. Cormack. A review of classification. *Journal of the Royal Statistical Society*, A 134:321–367, 1971.

[2] Trevor F. Cox and Micheal A. Cox. *Multidimensional Scaling.* Monographs on

Statistics and Appied Probability. Chapman and Hall, 1994.

[3] H.S.M. Coxeter. *Non-Euclidean Geometry.* University of Toronto Press, 1957.

[4] J. deLeeuw and I. Stoop. An upper bound for SSTRESS. *Psychometrika*, 51:149–153, 1986.

[5] Hansjörg Klock and Joachim Buhmann. Multidimensional scaling by deterministic annealing. In *Proc EMMCVPR Venice*, 1997.

[6] J. Lamping, R. Rao, and P. Pirolli. A focus+context technique based on hyperbolic geometry for viewing large hierarchies. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 401–408, 1995.

[7] John Lamping and Ramana Rao. Laying out and visualizing large trees using a hyperbolic space. In *ACM Symposium on User Interface Software and Technology*, pages 13–14, 1994.

[8] Frank Morgan. *Riemannian Geometry: A Beginner's Guide.* Jones and Bartlett Publishers, 1993.

[9] Tamara Munzner. H3: Laying out large directed graphs in 3d hyperbolic space. In *Proceedings of the 1997 IEEE Symposium on Information Visualization, Phoenix, AZ*, pages 2–10, 1997.

[10] Tamara Munzner. Drawing large graphs with h3viewer and site manager. In *Proceedings of Graph Drawing '98, Montreal, Canada, Springer-Verlag*, Lecture Notes in Computer Science 1547, pages 384–393, 1998.

[11] Jörg Ontrup and Helge Ritter. Text categorization and semantic browsing with self-organizing maps on non-euclidean spaces. In *Proc PKDD-2001*, pages 338–349. Springer LNAI 2168, 2001.

[12] Peter Pirolli, Stuart K. Card, and Mija M. Van Der Wege. Visual information foraging in a focus + context visualization. In *CHI*, pages 506–513, 2001.

[13] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling. *Numerical Recipes in C – the Art of Scientific Computing.* Cambridge Univ. Press, 1988.

[14] Kirsten Risden, Mary P. Czerwinski, Tamara Munzner, and Daniel B. Cook. An initial examination of ease of use for 2d and 3d information visualizations of web content. *International Journal of Human Computer Studies*, 53(5):695–714, 2000.

[15] H. Ritter. Self-organizing maps on non-euclidean spaces. In S. Oja, E. & Kaski, editor, *Kohonen Maps*, pages 97–110. Elsevier, Amsterdam, 1999.

[16] Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 5(24):513–523, 1988.

[17] J. W. Sammon, Jr. A non-linear mapping for data structure analysis. *IEEE Transactions on Computers*, 18:401–409, 1969.

[18] J.A. Thorpe. *Elementary Topics in Differential Geometry.* Springer Verlag, 1979.

[19] William P. Thurston and Jeffrey R. Weeks. The mathematics of three-dimensional manifolds. *Scientific American*, July:94–107, 1984.

[20] Jörg Walter and Helge Ritter. On interactive visualization of high-dimensional data using the hyperbolic plane. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 123–131. SigKDD, August, Edmonton, Canada 2002.