Spherical Similarity Explorer for Comparative Case Analysis

Leishi Zhang[1], Chris Rooney[1], Lev Nachmanson[2], William Wong[1], Bum Chul Kwon[3], Florian Stoffel[3], Michael Hund[3], Nadeem Qazi[1], Uchit Singh[1], and Daniel Keim[3]
[1] Middlesex University, London, UK
[2] Microsoft Research, Redmond, USA
[3] University of Konstanz, Konstanz, Germany

Abstract

Comparative Case Analysis (CCA) is an important tool for criminal investigation and crime theory extraction. It analyzes the commonalities and differences between a collection of crime reports in order to understand crime patterns and identify abnormal cases. A big challenge of CCA is the data processing and exploration. Traditional manual approach can no longer cope with the increasing volume and complexity of the data. In this paper we introduce a novel visual analytics system, Spherical Similarity Explorer (SSE) that automates the data processing process and provides interactive visualizations to support the data exploration. We illustrate the use of the system with uses cases that involve real world application data and evaluate the system with criminal intelligence analysts.

1. Introduction

Comparative Case Analysis (CCA) is a widely used technique for criminal intelligence analysis [15]. Given a collection of crime reports that contain both structured fields such as crime type and unstructured fields such as modus operandi, CCA aims at analyzing the commonalities and differences between them for predicting criminal activities, determining current, new or emerging problems and highlighting prevention, reduction or diversion opportunities [31]. Currently, CCA is a manual process. After reading the crime reports, the analyst identifies relevant headings (factors) that are considered to be useful for their understanding of the cases. Information from the reports is then collated under the headings for comparative analysis, result in a CCA table, where the analyst scans across crime rows to identify (dis)similarities.

With the increasing size of data and variety of information that is collected presently, the manual process becomes increasingly infeasible. First of all, identifying meaningful factors from large amount of data and collating corresponding information to form a CCA table is labor intensive and error-prone. Secondly comparison across tables with large amount of rows (crime reports) and columns (factors) is difficult. Analysts are looking for tools that can help them to extract relevant features from the data, and to visualize the data in graphical representations such that comparison can be made easy [4]. Software systems such as *IBM i2* [1] have been developed to assist the analyst with extracting information from large amounts of criminal data and link related information. But tools that support automatic data processing and similarity analysis are lacking.

We see parallels between CCA and traditional document analysis where structure needs to be extracted from unstructured textual fields. Therefore, we envision an automated CCA workflow starting with the extraction of meaningful features and concepts from the document collection. Subsequently, a CCA table is compiled by combining the information of the extracted features and concepts with existing structured fields in the data. The (dis)similarities between documents are then computed using a predefined distance measure that take into consideration values in both structured fields in the data and information of extracted features. Finally, the (dis)similarities are visualized in graphical representations to help the analyst identify patterns and relations in the data. These visual representations can then be presented in an interactive graphical user interface where the analyst can interact with the data and look at the data from different perspectives.

In this paper we propose a visual analytics pipeline that advances existing CCA approach by automating the CCA table generation process, computing the (dis)similarity between documents, and visualizing the (dis)similarity and the data in an interactive visual display to support reasoning and sense making. We implement the pipeline in *Spherical Similarity Explorer* (SSE), a visual analytics tool that integrates a spherical embedding method with a series of analysis, visualization and interaction techniques to support Comparative Case Analysis (see Fig. 1).

The contributions of this paper include 1) a visual analytics pipeline for CCA; and 2) a novel interactive visualization technique that maps textual documents to a 3D spherical surface and allows interactive exploration of the similarity between documents. The remainder of this paper is organized as follows: Section 2 introduces the research background; Section 3 discusses related work; Section 4 introduces the visual analytics pipeline of our *SSE* tool, as well as the detailed embedding, interaction and visualization techniques; Section 5 demonstrates two use cases with real-world criminal data and presents feedback from end users; and Section 6 draws conclusions and discusses future work.

2. Background

The work proposed in this paper is motivated by challenges in CCA as part of the mission of the EU FP7 funded project *"Visual Analytics for Sense-making and Criminal Intelligence Analysis"* [4]. The project looks into the development of VA solutions that supports evidential reasoning and sense-making from a large volume of criminal intelligence data.

CCA is also called Similar Fact Analysis (SFA) [5]. The idea is to analyze similar features (factors) in different crime cases in order to support *criminal investigation* and *crime theory extraction*. The former aims at identifying similar crime cases that are committed by the same offender(s) or criminal organizations, and the latter aims at identifying the causes of certain types of crime for future crime reduction and prevention.

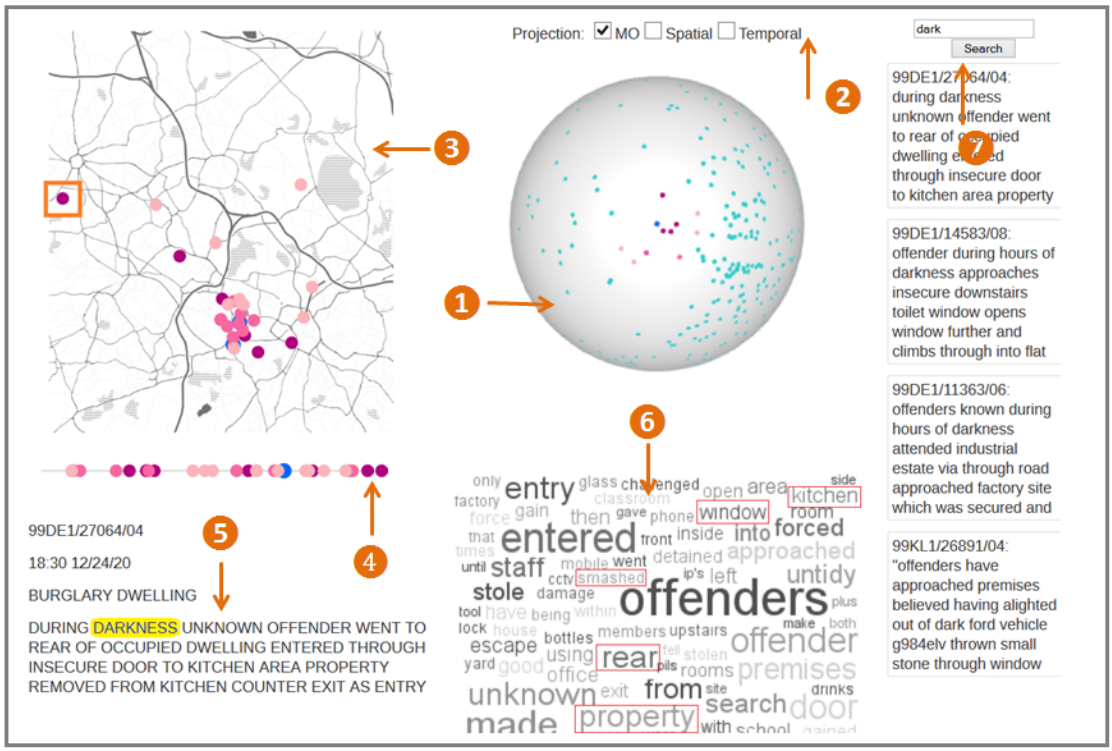

Figure 1. *Spherical Similarity Explorer: ❶ The Crime reports are mapped to an interactive spherical surface according to their (dis)similarities. ❷ The user can decide which features to include for projection. ❸ The user can select a document on the spherical surface and see similar documents highlighted in purple, the density of the color indicates the similarity level, higher density indicates higher similarity score. Close neighbors are projected onto a Geo-map. The point in the orange rectangular frame is an example of a very similar crime that happened at a distant location. ❹ The crimes are also projected to a time line. ❺ The details of the selected document ❻ The word cloud visualization allows the user to analyze the common features of a set of documents; in this case, terms such as "rear", "property", "smashed", "kitchen" and "window" occur frequently in the crime reports; these terms may be used to generate hypothesis about the crime pattern. ❼ The keyword search allows the user to investigate documents that contain the given keywords. The user can click on a result item to see its projection on the sphere.*

CCA involves data exploration. For example, new clues may be found while viewing key features of a particular crime. In that case analysts often want to view details of other similar crimes. This key feature identification is an iterative and incremental process. Consequently the analyst's focus of interest often moves from one case to another. A convenient visualization tool would support such exploration by moving the "point of interest" to the center of the drawing canvas such that the analyst can pay closer look into the details of the data point and examine the neighborhood easily.

One of the most commonly used visualization techniques for displaying (dis)similarities within large document collections is the so called "galaxy view", as it is for example implemented in IN-SPIRE [2], a state-of-the-art visual document analysis software. In the "galaxy view" each document is represented by a point, and the dissimilarities between documents are represented as Euclidean distances between points. By placing similar object close to each other and dissimilar object far apart, the visualization reveals patterns such as groups and outliers in the data.

The "galaxy view" helps the analyst to see patterns and relations in the data. But the method lacks flexibility in terms of navigation. Once a visual embedding has been generated, each object has a fixed location in the view. When the analyst's focus of interest moves from one document to another, it is hard to adjust the "point of interest" to the center of the display accordingly. Moving an arbitrary point in the display to the center requires either moving the drawing canvas, which may cause loss of information, or re-generating the layout, which may destroy the existing mental map of the analyst and demand additional computation. A"fish-eye" view [21] may be applied to enlarge the area of interest and at the same time preserve the mental map of the viewer, but every time the point of interest moves, a new layout needs to be computed.

Exploring the world map on a globe is an intuitive way of understanding distances and relations between different locations. The metaphor can be effectively adopted to support explorative analysis of distances and relations between objects in HD data. In the past various visual embedding methods have been proposed to map HD data to a 3D spherical surface [8, 10, 17, 27, 45], but little work has been reported on integrating them to VA tools for visual data exploration. This is hardly surprising, because without effective interactions and detail-on-demand visualizations, the method cannot fully support explorative analysis.

In *SSE* we utilize the advantage of the 3D globe metaphor by integrating a spherical embedding method with a series of interactions and linked-view-visualizations. At the back-end we develop

a series of text analytics methods to extract features and concepts from the text and compute the (dis)similarity between documents. The aim is to support interactive similarity analysis and data exploration required by CCA. The proposed approach is not limited to the *Spherical MDS* [17] that is currently implemented in the system – it can be easily substituted by any other embedding method that generates spherical embeddings. And the interactive spherical projection is also not limited to textual data – it can be adapted easily for analyzing multidimensional data of other data types.

3. Related Work

In this section, we review previous studies that motivate and inspire our approach.

3.1 Comparative Case Analysis

CCA is based on the notion of "comparison", which is a fundamental tool of analysis and widely used by many social sciences and scientific domains [14]. CCA starts with data preprocessing. Typically the data contains information gathered from different sources including police reports, witness statements, etc. and is stored in a semi-structured manner. A crime record may contain structured fields such as location, time, and type of crime, as well as unstructured textual fields such as modus operandi (abbreviated as MO) and intelligence notes [15].

A challenge of CCA is the feature extraction and similarity computation. Most of the feature extraction work reported in the literature is done by hand. For example, Bennell et al. [7] manually extracted different types of behavioral features such as entry behavior (front door, back door, climb to second floor etc.), target selection choices (petrol pump), property stolen (jewelery) and offender's spatial behavior as linking features to compute the pairwise similarity between crime cases. These features are extracted from MO of 86 solved commercial burglaries committed by 43 serial offenders.

It is not difficult to imagine the significant amount of work that is involved in the aforementioned process. Nowadays the data CCA has to handle often contains thousands of records or more. The task cannot be carried out without the help of automated feature extraction techniques. *SSE* tackles the challenge by designing effective entity extraction and concept extraction techniques that automatically extract important features from the data and compute corresponding measures. The technical details can be found in Section 4.1.

Bennell et al. [7] used the dichotomously coded values of the above mentioned linking features to examine if high degree of similarity between them enables different cases to be validly linked to a common offender. The similarity measure was computed on the basis of *Jaccard coefficients* between pairwise crimes. Their approach was twofold, first behavioral features were identified using logistic regression analysis capable of distinguishing between related and unrelated crime pairs. Secondly, Receiver Operating Characteristic (ROC) analysis was performed to assign each behavioral feature an overall predictive score.

Canter [13] investigated the consistency of features across organized and disorganized cases based on content analysis and *Multidimensional Scaling (MDS)*. His research revealed that disorganized features were either easy to identify or more common, probably due to their vast number compared to organized features. *Jaccard coefficient* was used to measure the proportion of co-occurring features. Given that CCA data contains both textual and non-textual fields, a simple metric that measures the textual features can miss important clues in the comparison such as time and location of a crime. In *SSE* we enhance the similarity computation by designing a composite measure that measures geospatial, temporal and textual features separately and merges them together for comparison. Details of the method can be found in Section 4.2.

3.2 Visual Embedding of Multi-dimensional Data

Mapping HD data to a lower-dimensional (LD) visual display is a difficult problem that has significant research interest. The main objective is to generate a meaningful LD (visual) representation of data in HD space such that people can gain understanding of the structure of the data, as well as the relationship dis(similarity) between data items. The process of representing HD data in a LD space such as a sheet of paper or computer screen is referred to with various terms or expressions. A general name for this is *embedding*. In the specific case of Cartesian data, *dimensionality reduction*, *projection*, *mapping*, and *manifold learning* are commonly encountered.

The key characteristic of any embedding technique is its underlying model, namely, the way it transforms data. This coordinate transformation can be linear or nonlinear. It can also be parametric or non-parametric. Parametric models provide out-of-sample extensions for free, that is, the transformation applies to new data points. The output of nonparametric methods is not really a model or transformation, but rather the transformed data. As the transformation remains implicit, the generalization to new data points is not straightforward.

The generic idea hidden behind all embedding techniques is that they should produce LD representations that preserve meaningful structural properties of data. In general, these properties formalize proximity relationships. They can be similarities (adjacencies, dot products) or dissimilarities (distances, angles).

The ancestor of all embedding methods is undoubtedly PCA [33]. The method can be interpreted in various ways (maximal variance preservation, minimal encoding/decoding error, total least squares, variable decorrelation). It can be carried out with various spectral decompositions (eigenvalues and singular values) or neural algorithms [32]. PCA is a linear projection technique. Refined cost functions have led to many variants, such as projection pursuit [25]. It is noteworthy that PCA yields exactly the same results as classical metric multidimensional scaling (MDS) [46]. Classical MDS is actually the dual form of PCA, in which the covariance matrix is replaced with the centered Gram matrix of dot products. Hence, classical MDS (and PCA) optimally preserve dot products.

More recent forms of MDS rather implement the closely related principle of distance preservation. Various cost functions (often called *s*tress) allow a wide variety of methods to emerge [16]. The most famous is probably Sammon's nonlinear mapping [36]. Like many MDS variants, the stress optimization with an elaborate gradient descent technique produces nonlinear embeddings. Yet another variant is curvilinear component analysis (CCA) [19], which behaves in a radically different way as other MDS-like techniques.

Quite naturally, the input data for MDS often comprise Eu-

clidean distances, which are easily converted into the corresponding dot products. Actually this is not mandatory at all and other types of distances can be used as well, such as the so called geodesic distances [40, 26]. The transformation of the reference distances in the stress function can also be identified in an optimization process. This is the principle of nonmetric MDS [38, 23]: a nonparametric and monotonic distance transformation is identified with either isotonic optimization or semidefinite programming [44].

Dot products or distances are not the only structural features that are useful in embedding techniques. Recent successful methods are based on similarity preservation, such as stochastic neighbor embedding (SNE) [22] and its variants [41, 42]. Yet another possibility close to the world of artificial neural networks is the well-known self-organizing map [43]. Though the SOM is not an embedding technique in the usual sense, it remains a very powerful visualization tool. The idea is similar to the one of PCA: fit a plane within the data cloud. In this case the plane is a kind of articulated and elastic grid, which can deform itself to reproduce curved shapes and clusters. This shows that the SOM works the other way round, compared to most embedding techniques. Instead of embedding data into a LD space, a predefined two-dimensional grid is fitted in the HD space. The heuristic algorithm of the SOM has been reformulated in more principled methods, such as the generative topographic mapping [9], which relies on an expectation-maximization procedure.

The review of the state of the art would not be complete without mentioning spectral embeddings. This family of methods shares a common framework: they apply classical MDS in a feature space, that is, nonlinearly transformed data variables. This framework extends classical MDS to nonlinear embeddings while keeping most of its advantages, like computational simplicity and convex optimization. The oldest method in the family is kernel PCA [37].

Isomap [40] is classical MDS applied to geodesic distances instead of Euclidean ones. Assuming that data are sampled from a manifold, the shortest paths in a neighborhood graph are computed in order to approximate the manifold geodesics. Using geodesic distances instead of Euclidean ones in classical MDS is expected to lead to better unfolding and hence to better embedding [26]. Another well known spectral method is locally linear embedding (LLE) [34]. The idea here is to first determine weights in order to reconstruct each data point from a mixture of its neighbors and secondly to use these weights to compute a low-dimensional representation of all points.

3.3 Spherical-based Visualization

Spherical visualization is a commonly used visualization technique for displaying relationships between objects, especially when the objects come with their geographical locations on Earth, or their positions in space [28]. If the geographical locations are not available or, as in our case, is not the sole attribute, spherical visualization maintains some useful features when compared to rendering on the plane. For example, some systems [29, 30] map the objects to the sphere to achieve focus+context distortion.

Mapping objects to a spherical surface instead of a 2D plane has several advantages. First of all, unlike a 2D plane, the spherical surface does not have a boundary, which provides an intuitive Gestalt association of relationships between nearby objects [11]. Secondly the spherical mapping helps avoiding the "corner effect" of 2D mappings [12] although projection from HD data space to a LD space will potentially result in distortions [6]. When the number of points is large, the method also lessens the over-plotting problem by having a larger projection space (see Fig.3). Furthermore, the design aligns with the "Focus + context" principle [39] that hides distant data points without losing context.

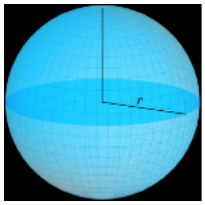

Figure 3. *Spherical surface - given that the area of a squared 2D visual display ($width = length = 2r$) is $4r^2$, the surface of an interactive sphere that can be fit in the display is $4\pi r^2$.*

The methods described in Section 3.2 can be applied to generate 2D or 3D embeddings of multi-dimensional data. In this paper we want to generate embeddings that lie on the surface of a sphere. A 3D spherical mapping of multi-dimensional data can be achieved in many ways. In theory any Dimenionlity Reduction (DR) method can be extended to fit the purpose by adding constraints. For example, a "dummy point" that has the same distance to all the other points can be added to a MDS approach to achieve a circular or spherical embedding [10, 8, 27]. Also, methods were developed to generate spherical embeddings using a SOM [45].

In *SSE* we implement the Cox and Cox's Spherical MDS approach [17]. Compared to other spherical embedding approaches that requires the generation of an artificial "dummy point" [10], [8] and [27] or tuning of parameters[45], Cox and Cox's method achieves non-metric MDS on a sphere in a simpler manner. *SSE* extends the capability of Cox and Cox's method by integrating it with a series of interactions and detail-on-demand visualizations, and embedding the spherical view in a multiple-coordinated-view where the analyst can examine the data from different perspectives. Details of the methods can be found in the section 4.3.

4. Visual Analytics Pipeline

Fig. 2 illustrates the visual analytics pipeline of *SSE*: it takes crime and incident reports as input, processes the reports by extracting relevant features and concepts from the reports, combines them with the headings of structured fields to form headings of a CCA table. Once the headings are generated, the system will check through each crime report to fill in the corresponding values in the CCA table. These values are used to compute the pairwise similarities between crime cases, resulting in a distance (dissimilarity) matrix. The distance matrix is then fed into the *spherical MDS* algorithm that generates spherical coordinates for each of the crime reports. The coordinates are used to map the crime reports to the spherical surface. The 3D sphere is interactive; the user can move the sphere for data exploration. Detail-on-demand analysis such as hot spot identification and visualization, and common feature analysis and visualization are also supported by the system.

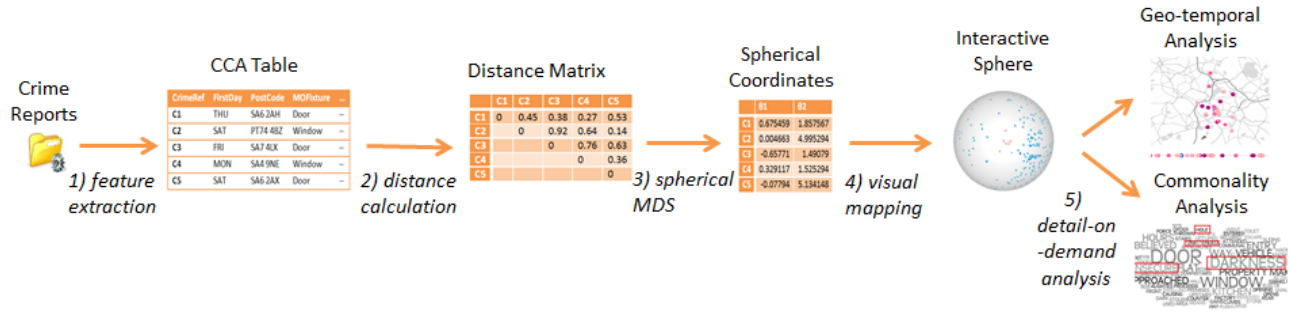

Figure 2. *The Visual Analytics Pipeline for Comparative Case Analysis (CCA): 1) Given a collection of crime reports our approach starts with extracting important features from the data, resulting in a CCA table where each record encodes the key feature values of a crime record; 2) similarities (distances) between crime records are computed using a composite distance measure, resulting in a distance matrix; 3) the distance matrix is then fed into a spherical MDS algorithm that computes the spherical coordinates for each crime record on the 3D spherical surface; 4) the coordinates are used to map crime records to the surface of the sphere; 5) Detail-on-demand analysis and visualization: (top) hot spots extraction and visualization, (bottom) common feature extraction and visualization.*

4.1 Feature Extraction

The features extracted from the crime reports are based on predefined patterns built to capture sequences like *unsecured premises* or *entered through rear door*. Compared with bag-of-word stemming features computed from documents [35], the extracted features are characteristic for the analyzed description of the crime procedure. This approach reflects the fact that two crime reports referring to offenders entering through a rear door at night are more similar to each other than to a report omitting the time of day. The patterns generating the features and sequences from the text are matched to a lemmatized version of the crime reports so not to miss any matches because of word inflections. The result is organized in groups of concepts, which are predefined by experts from the field of the analyzed data. These concepts can be parts of a building, makes and models of vehicles, and more. For example, having crime reports dealing with burglary, the concepts contain parts of buildings, expressions for the time of the day, and typically-used tools. This expert knowledge makes sure that the extracted features are relevant to the analyzed data and carry important information which not only the machine but also the analyst understands. In addition, the features represent chunks of the analyzed crime report, can be interpreted by a machine as well as humans, and make the information extraction process transparent to an analyst working with the generated data.

The extraction process is based on four main components: 1. *Preprocessing*: the reports are normalized, a cleanup removes all characters which do not carry any information (non printable and control characters). Based on a dictionary, abbreviations are expanded. 2. *POS Tagging*: a POS tagger determines the part of speech of each token in the report. 3. *Lemmatization*: a rule-based lemmatizer computes the lemma of each token. 4. *Pattern Matcher*: the pattern matcher finds occurrences of the predefined patterns (including permutations). It extracts the text matches, and generates the corresponding feature vector per crime report. The resulting feature vector is similar to a classical term feature vector. Instead of single word terms, it keeps track of the occurrences of multi-word terms extracted from the predefined patterns.

4.2 Distance Computation

When computing the dissimilarity between two crime reports we apply a so-called *composite distance measure* to handle the heterogeneous structure of the reports. As mentioned above, each report is described by a feature vector consisting of different data types such as time and geographically related information of the incident as well as extracted concepts of the modus operandi field. Our distance measure combines the different feature types into a single similarity value.

To combine feature types a *feature type dependent* distance function is applied to every considered feature. For time-related features, the similarity is measured by the time difference, for geo-related features the geographical distance between two locations is considered, and for the modus operandi field the distance is computed by counting the number of common concepts. More formally, the distance between the concepts $\{c_0,\ldots,c_{n-1}\}$ of the modus operandi MO_m and MO_n fields are computed as follows:

$$dist(MO_m,MO_n) = \frac{1}{\#concepts} \cdot \sum_{i=0}^{n-1} dist(MO_{m_{c_i}},MO_{n_{c_i}})$$

where $\#concepts$ refers to the number of non-missing concepts and

$$dist(MO_{m_{c_i}},MO_{n_{c_i}}) = \begin{cases} 0 & if\ m_{c_i} = n_{c_i} \\ 1 & else \end{cases}$$

As indicated above, a concept is ignored if both crime records have a missing value.

After computing and normalizing the distances between the separate features, the resulting distance between two crime records is computed by averaging the individual distances. As indicated in Fig. 2, the result of the distance computation step is a matrix containing all pair-wise distances between all crime reports.

4.3 Spherical MDS

SSE implements Cox and Cox's *Spherical MDS* algorithm [17] for mapping the distance between documents to the visual display. The algorithm adapts the original non-metric MDS

of Kruskal [24] to give a configuration of points that lie on the surface of a sphere.

Given a set of n objects under consideration and the dissimilarity between pairs of objects in the multidimensional data space given by $\{\delta_{ij}\}$, and the Euclidean distance between corresponding points in the lower dimensional embedding given by $\{d_{ij}\}$, Kruskal's non-metric MDS tries to "best" approximate the distance between objects in the data space to the embedding space by minimizing the cost function

$$S = \sqrt{\frac{\sum(d_{ij} - \hat{d}_{ij})^2}{\sum d_{ij}^2}}$$

where $\hat{d}_{ij}$ is the primary monotone least squares regression of $\{d_{ij}\}$ on $\{\delta_{ij}\}$.

Spherical MDS maps points onto a surface of a unit sphere. Each point i on a surface of the sphere is determined by two angles θ_i and ϕ_i (see Fig. 4). Assuming $-\frac{\pi}{2} \leq \theta_i \leq \frac{\pi}{2}$ and $0 \leq \phi_i \leq 2\pi$ for the spherical angles, the point i has the Cartesian coordinates

$$x_i = \cos\theta_i \sin\phi_i \quad (1)$$

$$y_i = \sin\theta_i \sin\phi_i \quad (2)$$

$$z_i = \cos\phi_i \quad (3)$$

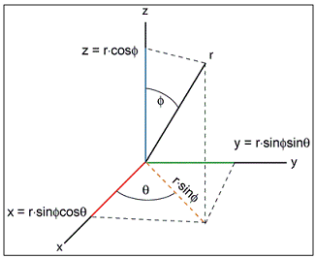

Figure 4. *Spherical Coordinates*

The distance between two points is defined as the shortest arc length along the great circle which passes through the two points (geodesic distance on the sphere).

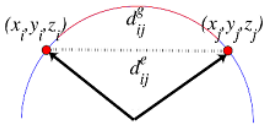

Figure 5. *Geodesic distance between two points on a sphere*

The Euclidean distance d_{ij}^e between two points i and j on a sphere is given by

$$d_{ij}^e = (x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2$$

substituting Eqs. 1-3, we have

$$d_{ij}^{e\,2} = 2 - 2\sin\phi_i \sin\phi_j \cos(\theta_i - \theta_j) - 2\cos(\phi_i)\cos\phi_j$$

By applying the law of cosine to the triangle depicted in Fig. 5, where φ is the planar angle for the unit sphere $r = 1$

$$d_{ij}^{e\,2} = 2r^2 - 2r\cos\varphi$$

the geodesic distance on the sphere d_{ij}^g is given by

$$d_{ij}^g = \arccos\frac{2 - d_{ij}^{e\,2}}{2}$$

Starting from an initial configuration of points (which is generated by a metric MDS), a configuration giving minimal stress can be found using the method of steepest decent. As Cox and Cox pointed out, since there is a one-to-one increasing relationship between the Euclidean distance and the arc length (geodesic distance) either of them can be used for the stress minimization. The resulting MDS solution will not be affected.

4.4 Interactions

While a variety of 3D interaction devices are available to interact with 3D visualizations [18], we wanted our application to be device-independent and work both with and without novel hardware, making the application more accessible for those with standard interaction devices. Using simple, cursor-based interaction, the sphere can be rotated in both the y- and x-axis by clicking and dragging the cursor across the sphere. Scroll functionality, such as a two-finger swipe on a touch-pad, can be used to zoom in, revealing further relationships in dense regions. These gestures also lend themselves well to touch devices, where zoom is controlled using the de-facto pinch gesture.

We also investigated mid-air gestures through the use of a Leap Motion device [3]. Users could rotate and zoom into the sphere by holding their hand over the device and moving side-to-side and backwards-and-forwards respectively. After some initial novelty, we found the interaction to be clumsy. With no point of reference, it was difficult to rotate the sphere without accidentally zooming, it was also no surprise that fatigue soon set in. One gesture that did work well was the ability to change the opacity of the sphere by making a grabbing gesture.

Since the data points are small, and dense in some regions, we recommend cursor-based interaction, where we also have the affordance of hovering over data points on the sphere. This can be used to reveal details of a crime, such as showing *when* and *where* it occurred on the geographic and temporal visualizations, and showing a summary of the crime details, including the offender's modus operandi. Clicking on a crime highlights its nearest neighbors, using a three-tier color scale to group projected distance. These points are also placed on the geo-spatial and temporal views to allow for potential new relationships to be discovered.

We intend the smooth and fluid interaction with the sphere to encourage playful exploration, discovering new and unexpected relationships in a serendipitous manner. We combine this with smooth animations that allow users to view data items, selected through the search and retrieval interface, at the front-center by rotating the sphere into position. This helps the user to maintain context as they move from one data item to the next. We also use animation to move data points to their new locations when the projection is changed.

4.5 Visualizations for Similarity Analysis

We design the visualization components of *SSE* in such a way that different types of analysis can be performed through coordinated views. According to the National Policing Improvement Agency [31] there are four main types of Crime Pattern

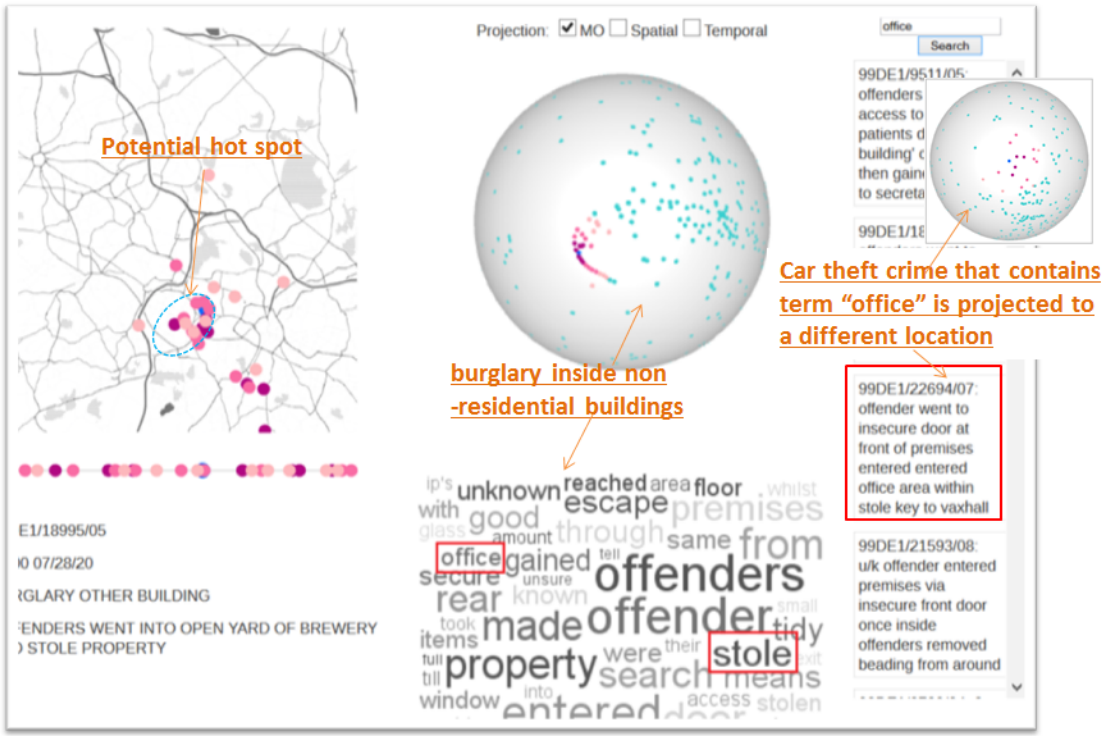

Figure 6. *Similarity Analysis based on Modus Operandi: the pattern shows a group of crimes that are related to theft inside non-residential dwellings such as offices and school buildings. The geo-map highlights a potential hot spot of the crime type. The time line visualization indicates this type of crime cases is evenly distributed over time. The third crime record of the search query contains the term "office" but relates to a car crime that is projected to a distant location.*

Analysis: *hot spot identification*, *crime trend identification*, *crime series identification* and *general profile analysis*. We use these tasks to drive the design of our different views:

3D Sphere: The spherical embedding provides an overview of the similarities between crime cases. It is useful in understanding global patterns in the data such as groups and outliers. The user can explore the data by various interactions as introduced in Section 4.4. It also allows the user to select a particular document on the sphere and see the detailed modus operandi. Similar documents will be highlighted in corresponding views.

Feature Selection: *SSE* implements a feature selection panel that allows the user to select different feature set combinations, namely MO (modus operandi), spatial (geo-spatial locations), and temporal (time and date) of the crimes. The similarities between crime cases are always computed based on selected features. By adding or removing a feature for similarity computation, the analyst can test different hypotheses.

Geo-spatial visualization: The system also includes a map view that shows the distribution of crimes on a geographical map. By projecting crimes to the map, the analyst can easily identify crime hot spots and drill down to individual crime cases that happened in a particular geographical region.

Time-line visualization: A time-line visualization is implemented to show the distribution of crimes over time for cross referencing between geo-spatial and temporal relations in the data.

Commonality Analysis: The user can select a group of documents and see the common features in a word cloud visualization. This provides an overview of the key terms in the data and helps the analyst to identify clues for reasoning and sense-making.

Search and Filtering: The search panel allows the analyst to perform keyword search. Details of the matching documents can be viewed in the panel below. The filtering helps the analyst to quickly focus on a subset of documents that are of interest.

Linked Views: all the views mentioned above are linked. Change in one view will trigger corresponding changes in other views. For example, selecting a document in the spherical surface will update the map view.

5. Use Cases and User Evaluation

In this section we present two use cases of the system and some feedback from the end user. We use 1588 anonymized real crime records as input data of the use cases and use them to demonstrate the functionality of the system to the end user. The data contains 27 structured fields recording various types of information about the crime plus a free text field that stores the modus operandi of each crime.

5.1 Similarity Analysis Based on Modus Operandi

First we use the features extracted from the modus operandi for similarity analysis. Extracted features from the modus operandi field include concept based features such as MO_Position (rare, front, size of the entry), MO_FixtureType (window, door, etc.), MO_fixtureMaterial (plastic, metal, etc.), and MO_search (tidy, untidy) as well as key terms contained in the modus operandi descriptions such as *alarmed*, *dog*, and *car-*

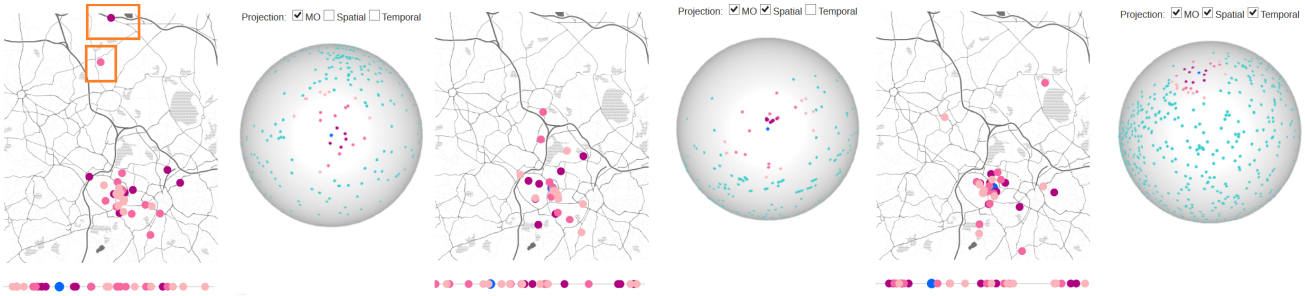

Figure 7. *Similarity analysis based on a different subset of features: given the same document, the neighborhood changes when different feature sets are selected for comparison.*

key.

Once the data has been processed and projected onto the spherical surface, one can easily see a "spoon" shaped pattern (cluster) (see Fig. 6). By clicking on one of the center points of the cluster and checking the word cloud visualization of the neighborhood crimes, it is not difficult to see that most of the crime cases occurred in non-residential dwelling such as offices, school buildings and commercial premises. The word cloud helps with identifying more commonalities between the crimes — terms *stole* and *office* frequently occur in these crime cases. This leads to the hypothesis that the pattern is related to burglary/theft inside non-residential dwellings. By double-checking the modus operandi descriptions of the cluster members, the hypothesis can be further confirmed.

The geographical map at the top left panel helps the analysts understand the geographical distribution of the crime cases. The small region at the southern part of the city looks like a hot spot of theft in commercial premises. This can lead to further investigation. For example, the link between the specific type of crime to the highway next to the area, the distribution of commercial vs. residential buildings in the area, or the temporal pattern of the crimes (office or non-office hour, week or week-end days), etc.

The analyst can also search for other crime cases that contain the keyword *office* and see whether there are other types of crime that are related to commercial premises. In this case, the third item in the query result is projected some distance away from the identified pattern (see Fig. 6). This is not surprising, because although the modus operandi does contain the term *premise*, the crime itself is a car crime that happened outside the building — the offender stole the car key in an office and drove the car away.

5.2 Similarity Analysis Based on a Different Subset of Features

Next we investigate the embeddings of the same data using different combinations of three feature sets that are implemented by the current version of the SSE tool: *MO features* (modus operandi), as described in the previous use case, *spatial features*, which includes the latitude and longitude of the crime location, and *temporal features*, which include the date and time when the crime happened. We randomly select a crime report (ID "99DE1/14452/04") from the data as an "anchor" point and search for similar cases using different feature set combinations. We look at 3 different combinations: MO, MO+Spatial, and MO+Spatial+Temporal.

Fig. 7 shows the resulting embeddings. Reasoning can be made based on some of the comparisons. For example, from the map view, the MO-only projection does not seem to have a strong geo-pattern. The crimes that are similar to the anchor document are at distant geographical locations. Adding spatial features does not result in significant changes in the spread of the region, apart from filtering out a few documents that happened up north (highlighted in the red rectangular frame).

After adding temporal features to the existing two feature sets, the analyst can identify a different set of similar documents. Most of the crimes that are similar to the anchor crime seem to have occurred within a much smaller region. By analyzing the commonalities between these document sets, further hypotheses can be generated. For example, these documents could be a set of similar burglary cases that happened in a park area during night, assuming the term "dark" or "darkness" appears in several documents, and one of the document contains "broken" and "lamp". The "broken lamp" could be a key cue to the cases. Identification of these types of causal relationships in the data is crucial to future crime prevention and reduction.

5.3 User Evaluation

In order to validate our design and assess the usability of the tool, an evaluation session was carried out with criminal intelligence analysts. Three groups participated in the evaluation: one from a UK police force (2 analysts), one from a Belgium federal police force (3 analysts) and one from a Belgium local police force (2 analysts). During the session the tool was introduced to each group separately alongside 11 other prototype tools that have been developed for the project for different analysis tasks.

The evaluation session was interactive. A demo of the tool is given in front of the end user group. The analysts were encouraged to ask questions and provide feedback during the demo session. At the end of the session each analyst had to complete a questionnaire independently. The questionnaire consists of three types of questions: *usability*, *analytical work*, and *satisfaction*.

The feedback was encouraging. The analysts welcomed the exploratory nature offered by the interactivity of the tool and were able to see its potential to help with the tasks of CCA. They thought the tool is easy to understand and use. They liked the multiple-linked visualizations that allows them to inspect data from different perspectives. The analyst also felt that the underlying similarity computation could be used effectively to detect when particular modus operandi traits reoccur. For example, the tool could alert the analyst that a particular trait occurred ten times in the last fortnight, or that a trait that was frequent last summer

is becoming frequent again.

One rather unexpected feedback was on the word cloud visualization. One may argue a list of frequently occurred words is a better way of summarizing a documents collection than a word cloud. The analysts said they prefer the latter. The word could visualization seems to be a natural tool for sense-making and reasoning – often when their eyes see a word cloud, their brain automatically start to piece words together to form stories.

The questionnaire result also looks promising. Among the 12 tools that have been evaluated, SSE received one of the highest overall score (4.25 out of 5) , as well as satisfaction (4.83 out of 5), expectation (4.56 out of 5) and evidential reasoning scores (4.61 out of 5).

The analysts could also foresee a number of ways in which the tool could be improved. For example, they would like the system to reveal more details of the underlying similarity computation and provide a summary of why two crimes are considered either similar or different. This enables analysts to validate the computation and could be accompanied by functionality for modifying and correcting the similarity measure. A simplified method for this is semantic interaction [20] where analysts use drag-and-drop to reposition data points on the surface of the sphere, thus adjusting the underlying similarity weightings.

6. Conclusion and Future Work

In this paper we introduced the *Spherical Similarity Explorer*, a visual analytics tool for Comparative Cases Analysis. The system addresses several challenges in analyzing criminal intelligence data, including entity extraction and concept generation from textual fields, combining both "soft " (features extracted from textual fields) and "hard " (features that exist in structured fields) information for similarity analysis, and providing effective visual representations of the data. A visual analytics pipeline is designed and developed to automate the CCA process. The pipeline incorporates a series of text analytics techniques and interactive visualizations to automate the manual CCA process and support visual data exploration. A novel visualization technique that integrates the *spherical MDS* approach with interactions to support data exploration is proposed to avoid the over-plotting problem of 2D visual embeddings and to support flexible navigation. Various visualization and interaction techniques are integrated into the system to facilitate filtering, hot spot identification, and commonality analysis. Positive feedback was collected from police analysts, as well as further development directions

For future work we would like to improve our system according to some of the feedback from the end users. In particular, we intend to provide more transparency, flexibility and user control of the similarity computation, allow weighting on different features, highlight not only commonalities but also differences between crime cases, integrate semantic interactions with the system, and develop more filtering facilities to support "slicing and dicing" of the data before comparison.

In addition to the above mentioned functionality, we plan to extend the time line visualization to support more sophisticated temporal analysis. Visualizing the overlapping features between different sets of documents is another future direction that we would like to explore. We also plan to extend the spherical mapping by adding dynamic Azimuthal projections. This would allow the analyst to compare the absolute distances from one particular document to all the other documents on the spherical surface. Although the projection will need to be recomputed every time the analysis changes their focus, accurate mapping could be helpful when a criminal investigation focuses on a particular crime.

The tool presented in this paper is a system prototype as part of the *VALCRI* project. Our next goal is to improve the functionality, usability, scalability and robustness of the system and develop a fully functional software. Further user evaluations of the system will be conducted to guarantee the usability and effectiveness of the tool.

Acknowledgments

This work was supported by the EU project *Visual Analytics for Sense-making in Criminal Intelligence Analysis (VALCRI)* under grant number FP7-SEC-2013-608142. The authors would like to thank Dr Olaf Chitil for his help with code debugging and proof read of the paper.

References

[1] Ibm i2, http://www-01.ibm.com/software/uk/industry/i2software/, last retrieved 25/03/2015.

[2] In-spire, http://in-spire.pnnl.gov/, last retrieved 25/03/2015, 2015.

[3] Leap motion device, https://www.leapmotion.com/, last retrieved 25th march,2015.

[4] Visual analytics for sense-making and criminal intelligence analysis, http://www.valcri.org/, last retrieved 25th march,2015.

[5] *Working Manual of Criminal Law*. Carswell Legal Pubns, Mar 2000.

[6] M. Aupetit. Visualizing distortions and recovering topology in continuous projection techniques. *Neurocomputing*, 70(7-9):1304–1330, 2007.

[7] C. Bennell and D. V. Canter. Linking commercial burglaries by modus operandi: tests using regression and ROC analysis. *Science & Justice*, 42(3), 2002.

[8] P. M. Bentler and D. G. Weeks. Restricted multidimensional scaling models. *Journal of Mathematical Psychology*, 17:138–151, 1978.

[9] C. Bishop, M. Svensén, and C. Williams. GTM: A principled alternative to the self-organizing map. In M. Mozer, M. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems (NIPS 1996)*, volume 9, pages 354–360. MIT Press, Cambridge, MA, 1997.

[10] B. Bloxom. Constrained multidimensional scaling inn spaces. *Psychometrika*, 43(3):397–408, 1978.

[11] R. Brath and P. Macmurchy. Sphere-based information visualization: Challenges and benefits. In *Information Visualisation (IV), 2012 16th International Conference on*, pages 1–6, July 2012.

[12] A. Burenkov and J. Lorenz. Corner effect in double and triple gate finfets. In *European Solid-State Device Research, 2003. ESSDERC '03. 33rd Conference on*, pages 135–138, Sept 2003.

[13] D. V. Canter, L. J. Alison, E. Alison, and N. Wentink. The organized/disorganized typology of serial murder: Myth or model? *Psychology, Public Policy, and Law*, 10(3):293–320, September 2004.

[14] D. Collier. The comparative method. In *POLITICAL SCI-*

ENCE: THE STATE OF DISCIPLINE II, pages 105–118, 1993.

[15] N. Cope. Intelligence led policing or policing led intelligence?: Integrating volume crime analysis into policing. *Br. J. Criminol.*, 2004.

[16] T. Cox and M. Cox. *Multidimensional Scaling*. Chapman & Hall, London, 1995.

[17] T. F. Cox and M. A. Cox. Multidimensional scaling on a sphere. *Communications in Statistics - Theory and Methods*, 20(9):2943–2953, 1991.

[18] M. Csisinko and H. Kaufmann. Towards a universal implementation of 3d user interaction techniques. In *Mixed Reality User Interfaces: Specification, Authoring, Adaptation (MRUI'07)*, pages 17–24, 2007. Vortrag: IEEE Virtual Reality 2007, Charlotte, NC, USA; 2007-03-10 – 2007-03-14.

[19] P. Demartines and J. Hérault. Curvilinear component analysis: A self-organizing neural network for nonlinear mapping of data sets. *IEEE Transactions on Neural Networks*, 8(1):148–154, Jan. 1997.

[20] A. Endert, P. Fiaux, and C. North. Semantic interaction for visual text analytics. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 473–482, New York, NY, USA, 2012. ACM.

[21] G. W. Furnas. Generalized fisheye views. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '86, pages 16–23, New York, NY, USA, 1986. ACM.

[22] G. Hinton and S. Roweis. Stochastic neighbor embedding. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems (NIPS 2002)*, volume 15, pages 833–840. MIT Press, 2003.

[23] J. Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29:1–28, 1964.

[24] J. Kruskal. Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, 29(2):115–129, 1964.

[25] J. Kruskal. Toward a practical method which helps uncover the structure of a set of multivariate observations by finding the linear transformation which optimizes a new index of condensation. In R. Milton and J. Nelder, editors, *Statistical Computation*. Academic Press, New York, 1969.

[26] J. Lee and M. Verleysen. Curvilinear distance analysis versus isomap. *Neurocomputing*, 57:49–76, Mar. 2004.

[27] S.-Y. Lee and P. M. Bentler. Functional relations in multidimensional scaling. *British Journal of Mathematical and Statistical Psychology*, 33(2):142–150, 1980.

[28] P. Leong and S. Carlile. Methods for spherical data analysis and visualization. *Journal of neuroscience methods*, 80(2):191–200, 1998.

[29] T. Munzner. H3: Laying out large directed graphs in 3d hyperbolic space. In *Information Visualization, 1997. Proceedings., IEEE Symposium on*, pages 2–10. IEEE, 1997.

[30] L. Nigay and F. Vernier. Design method of interaction techniques for large information spaces. In *Proceedings of the working conference on Advanced visual interfaces*, pages 37–46. ACM, 1998.

[31] NPIA. National policing improvement agency: Professional practice on analysis, 2008.

[32] E. Oja. Principal components, minor components, and linear neural networks. *Neural Networks*, 5:927–935, 1992.

[33] K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2:559–572, 1901.

[34] S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

[35] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620, 1975.

[36] J. Sammon. A nonlinear mapping algorithm for data structure analysis. *IEEE Transactions on Computers*, CC-18(5):401–409, 1969.

[37] B. Schölkopf, A. Smola, and K.-R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10:1299–1319, 1998. Also available as technical report 44 at the Max Planck Institute for Biological Cybernetics, Tübingen, Germany, December 1996.

[38] R. Shepard. The analysis of proximities: Multidimensional scaling with an unknown distance function (parts 1 and 2). *Psychometrika*, 27:125–140, 219–249, 1962.

[39] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of the 1996 IEEE Symposium on Visual Languages*, VL '96, pages 336–, Washington, DC, USA, 1996. IEEE Computer Society.

[40] J. Tenenbaum, V. de Silva, and J. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, Dec. 2000.

[41] L. van der Maaten and G. Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

[42] J. Venna, J. Peltonen, K. Nybo, H. Aidos, and S. Kaski. Information retrieval perspective to nonlinear dimensionality reduction for data visualization. *Journal of Machine Learning Research*, 11:451–490, 2010.

[43] C. von der Malsburg. Self-organization of orientation sensitive cells in the striate cortex. *Kybernetik*, 14:85–100, 1973.

[44] K. Weinberger and L. Saul. Unsupervised learning of image manifolds by semidefinite programming. *International Journal of Computer Vision*, 70(1):77–90, 2006.

[45] Y. Wu and M. Takatsuka. Spherical self-organizing map using efficient indexed geodesic data structure. *Neural Networks*, 19(6-7):900–910, 2006.

[46] G. Young and A. Householder. Discussion of a set of points in terms of their mutual distances. *Psychometrika*, 3:19–22, 1938.