Visualization of Zoomable 2D Projections on the Web

Michael Maus[1], Tobias Ruppert[2], and Arjan Kuijper[1,2]([envelope])

[1] Technische Universität Darmstadt, Darmstadt, Germany
[2] Fraunhofer IGD, Darmstadt, Germany
arjan.kuijper@igd.fraunhofer.de

Abstract. The objective of the work is the research and development of a web-based visualization system for the creation and testing of zoomable projection cards. The basic idea is to project a multidimensional data set onto two dimensions using projection methods to represent it on a 2D surface. Based on the Card, Mackinlay, and Shneiderman visualization pipeline, a data processing model has been developed. For data processing various distance metrics, dimension reduction methods, zooming approaches as well as presentation concepts are considered. The peculiarities and considerations of the respective technology are discussed. A zooming approach allows large amounts of data to be displayed on a limited area. In order to better visualize connections within the data, concepts of presentation are discussed. The data points are represented as glyph-based objects or using color maps, various shapes, and sizes. Best practices about colormaps are discussed. In order to display large amounts of data in real time, a separation of the generation and visualization process takes place. During generation, a tabular file and selected configuration execute computationally-intensive transformation processes to create map material. Similar to Google Maps, the generated map material is represented by a visualization. Management concepts for managing various map sets as well as their generation and presentation are presented. A user interface can be used to create and visualize map material. The user uploads a tabular file into the system and chooses between different configuration parameters. Subsequently, this information is used to generate map material. The maps and various interaction options are provided in the visualization interface. Using various application examples, the advantages of this visualization system are presented.

Keywords: Visualization · Zooming · Projection

1 Introduction and Motivation

Getting an overview of a tabular dataset is still a challenge. In addition to the classic representation of the dataset in tabular form (such as Excel), visual exploration offers alternative solutions. Visual projection methods (Multi-dimensional

Scaling (MDS), Principal Component Analysis (PCA), etc.) make it possible to project multi-dimensional data sets on two dimensions, which are then displayed in 2D. Decisive advantages of visual projection methods are that relationships between the individual data sets are particularly well recognized. Relationships are expressed by distances based on topology-preserving approaches. Thus, "similar" data objects have a smaller distance from each other than "dissimilar" data objects, which have a large distance from each other and are therefore far from each other. In addition to the projection method, the choice of a suitable distance metric could influence the spatial arrangement of the objects. In addition, a heatmap for data visualization would be desirable. Heatmaps visualize the density of the object. A high density of objects exists when regions exist in which many objects are close to each other, whereas regions with few objects have a weak density of objects. Thanks to the different colors, the user is aware of the relationship with the object densities relatively quickly. Another challenge concerns the scalability of the visualization. Due to the limited display area, large amounts of data can hardly be displayed. The consequences would be overlapping data points, so that a clear assignment is impossible. Zoom-bare visual interfaces offer a solution for this. As the zoom level increases, the display area expands to allow more points to be displayed. Thus, large amounts of data could be represented by a suitable zoom approach. In addition, a client/server architecture with a web interface optimized for mobile devices would make sense. With the help of an Internet access, the visualization system of Anywhere (= client/server architecture) could be used to perform an exploratory data analysis. For optimization on mobile devices different display sizes, display resolutions have to be taken into account (= mobile first approach) [5]. Possible performance bottlenecks have to be overcome. The transfer of data between client and server, as well as the processing of the data on the mobile devices are critical processes that generate waiting times.

The aim of the work is the research and development of a web-based visualization system for the creation and testing of zoomable projection cards. The system should be based on the processing steps of the information visualization model of Card et al. [4]. Based on the model, the application should support the transformation process of the input data to map material, as well as their representation completely on the Web. In the app, the user selects a tabular text file. Panels set required configuration parameters. The user can manually set labels and features. Features are considered for the calculation of the X, Y coordinates. Labels describe characteristics that are displayed during a search. In addition, the user should choose between several similarity measures, projection method of his desired configuration. On the basis of a configuration and the file, map material is to be generated, which is displayed on request. Here, the configuration describes desired methods that are taken into account during processing. During processing, distances between the data objects are first calculated. With a selected dimension reduction method, the desired 2D data points are determined on the basis of the distances. The 2D data points are positioned on the map material. The map material is stored in the system and loaded on

demand. The problem of visual scalability should be solved by using a zooming concept. The objects should be faded in or out at different zoom levels based on a user-definable measure of interest. To provide map material as quickly as possible, a caching concept is developed. After the described transformation process, map sections are generated and saved as image material. At the request of a client, the server transmits the stored images "on-demand". A web interface, similar to Google Maps, displays the footage. Related approaches to the testing of projection results were published this year by Stahnke et al. presented [14]. A related work for creating spatiotemporal maps based on 2D tiles has been described by Lins et al. [10].

2   Concept

The goal of the visualization system is to visually represent large multidimensional data sets. Configuration parameters are to be set via a user interface, whereby the user can choose between different configurations. In addition, the approach should enable usability on different devices (mobility). The system must: Load and process tabular files and generate suitable visual representations. A concept has been developed that allows the above-mentioned interactive visualization pipeline. Related work has described various models and approaches that help to solve the following problems: For the transformation of the data into a visual representation, a pipeline based on the visualization pipeline of Card et al. [4] has been developed. To ensure visual scalability of the data, a semantic zoom approach [2,3] has been taken into account. In order to consider device independence as well as multi-user scenarios, a client-server architecture based on the considerations of the ZOIL framework [15] has been developed. Pre-calculated maps are generated to prevent complex projections from being made on each client request. Large map material is unsuitable for transmission to the client because of the high transmission time. With a tiling approach, the map material is divided into tiles (so-called subsections), which have a small fixed size and are therefore transmitted faster.

Full results can be found in [11].

2.1   Projection Pipeline

The goal is to visually represent a tabular multidimensional record so that the user can explore the relationships within the data in an exploratory way [9]. The user is provided with a graphical representation in which relationships between the individual objects are better recognized than in a tabular representation (as in Excel). Objects are the individual line entries of the table. Instead of a table, the user is provided with a 2D scatterplot that visualizes the similarities between the different objects. Thus, a procedure model is needed to transform a tabular file into a 2D scatterplot representation.

The information visualization pipeline from Card et al. [4] describes, at a high level of abstraction, a model for transforming data over several levels into

visual representations. Based on the Card et al. Visualization pipeline, we present a model that transforms multidimensional data. Starting from a data source, information is read, processed and visualized. The model performs 5 steps in succession. A data source reads the multidimensional data. From the multidimensional data, the relevant features are selected, with which a distance matrix is determined. The distance matrix contains all distances/distances between the individual objects. By using a projection method, the 2D data points are determined using the distance matrix and drawn on a scan plot. The following sections explain the considerations and special features involved in processing the individual steps.

2.2   Input

From a tabular file, multidimensional data is read and processed in further steps. The system must provide interfaces to read a tabular file. Within the scope of the work a calculation of the 2D data points using the projection methods is realized exclusively with numerical data. The focus is on numerical data that is interval scaled. A distance calculation of differently scaled data is limited possible. With similarity-based approaches, similarities between different strings or categorical data can be determined. Future work can take account of similarity-based approaches. Among other things, the pipeline is suitable for processing economic key figures as well as interval-scaled measurement data from research projects. The consideration of different data structures would be a meaningful extension and can be realized with relatively little effort. An extension would also allow the distance between hexcodes or dates to be determined.

For a suitable representation, relevant features must be identified and selected. Selecting features with no explanatory power will not recognize the context, or the result will be noisy. The choice of features can significantly affect the result of the projection. In order to avoid distortions or wrong connections, relevant features must be separated from irrelevant characteristics. For further processing within the pipeline, irrelevant features must be filtered. In order to determine suitable features, two strategies are suitable. In the first strategy, a simple model with few features is first set up, which is continuously expanded with additional features. In this procedure, characteristics for the model are successively selected. Irrelevant features distort the presentation and are therefore not included, whereas relevant features are included in the model and describe the desired context. Another approach is to include all variables except the non-numeric features in the model. In this procedure, individual characteristics are removed one after the other and a check is made as to whether an improvement is occurring. In the model, variables are removed until a desired relationship can be identified. In both strategies, as well as in general, an attempt is made to determine a compromise between overspecification by too many variables and a small explanatory power by too few variables.

2.3   Feature Selection

For a suitable representation, relevant features must be identified and selected. Selecting features with no explanatory power will not recognize the context, or the result will be noisy. The choice of features can significantly affect the result of the projection. In order to avoid distortions or wrong connections, relevant features must be separated from irrelevant characteristics. For further processing within the pipeline, irrelevant features must be filtered. In order to determine suitable features, two strategies are suitable. In the first strategy, a simple model with few features is first set up, which is continuously expanded with additional features. In this procedure, characteristics for the model are successively selected. Irrelevant features distort the presentation and are therefore not included, whereas relevant features are included in the model and describe the desired context. Another approach is to include all variables except the non-numeric features in the model. In this procedure, individual characteristics are removed one after the other and a check is made as to whether an improvement is occurring. In the model, variables are removed until a desired relationship can be identified. In both strategies, as well as in general, an attempt is made to determine a compromise between overspecification by too many variables and a small explanatory power by too few variables.

2.4   Distance Matrix

In order to map relationships between the objects, a suitable method is needed. Distances can be used to determine relationships between objects with numeric, metric-scaled features [13]. Metric scaling means that a ranking can be established and distances between the characteristic values can be measured. To achieve comparability between the various entries, normalization is required. Without a normalization, the entries would not be comparable, because the influence of certain features would be significantly larger than others. A min-max normalization unifies all features so that there are no size differences. When interpreting distances, it should be noted that distances with a higher value are farther apart and therefore less similar to one another, whereas a lower distance value describes a small distance between the objects and thus the objects are more similar to one another. Only numeric data are considered for the calculation. For a comparison of categorical data or strings special distance measures, also known as similarity measures, are needed, which are not part of this work. Supporting a variety of distance methods helps to create greater flexibility of graphical representations. Depending on the choice of the distance metric, it is possible in some cases for different representations to be generated. The distances between all objects are summarized in a distance matrix. Depending on the procedure, a distance matrix is optional or mandatory for the projection on 2D coordinates. In a PCA projection is often dispensed with a distance matrix, whereas in the MDS projection a distance matrix is mandatory.

2.5   Projection

Projection methods make it possible to use a given distance matrix to determine the desired 2D coordinates. In general, there is no central projection method that optimally explains all types or distributions of the data. Therefore, it is recommended to provide various projection methods in the visualization system.

PCA - also called *Principal Component Analysis* - describes a method for determining a linear projection of data that maps data to maximize variation in lower dimensional space. The goal is to explain the variation of the data with a small number of the most meaningful linear combinations ("main components"). The computation of the 2D coordinates does not necessarily require a distance matrix. Projection is also possible using the selected features of the multidimensional data set. The method is often used in linear relationships within the data. Thus, a linear relationship exists when the data is scattered around a straight line in the higher dimensional space. For non-linear relationships, the general PCA is inappropriate.

The *Kernel PCA* procedure describes an extension of the PCA, so that nonlinear relationships can be explained. It is useful if no meaningful representation of the data is generated with a PCA calculation. Using the Kernel PCA Trick, the input data is transformed using a selected function before the PCA calculation. Using the selected function, the data is transformed into a linear form and then the PCA projection is realized. With a Gaussian function, the data in the higher-dimensional space, which have a Gaussian distribution, can first be transformed into a linear form, in order then to carry out the PCA calculation. PCA, Kernel PCA Polynomial PCAs, and Kernel PCA Hellinger are unsuitable for imaging. Appropriate visualization is achieved using Kernel PCA Laplacian and Kernel PCA Gauss.

The *MDS* (*Multidimensional Scaling*) method describes a collection of analysis methods to represent objects based on their similarities in lower dimensional space. In contrast to the PCA method, a distance matrix is required. The MDS differentiates between ClassicalScaling and DistanceScaling. The Classical Scaling determines the 2D coordinates considering a distance matrix, using complex linear algebra. The distance matrix is transformed before the dimension reduction and then the desired coordinates are determined using a PCA projection. Distance Scaling uses an iterative approach and approaches a suitable representation in each iteration. First, the objects are randomly distributed in space and aligned in multiple iterations until an error threshold is reached. Objects that are more similar to each other are pushed toward each other, whereas objects that are more dissimilar are pushed away from each other.

The MDS methods are suitable for displaying objects based on their distances in the lower dimensional space. However, there are problems. As the number of entries increases, the distance matrix grows quadratically. From a size of more than 3000 entries memory leaks can occur. The required amount of data exceeds the main memory and errors occur. The ClassicalScaling should be preferred to DistanceScaling for metric scaled data. The DistanceScaling does not provide a consistent result, because with each new execution, the objects are randomly

distributed in space. As a result, there is no globally uniform arrangement, but a local optimal arrangement. Due to the multiple iterations DistanceScaling offers a weaker performance compared to ClassicalScaling.

FastMap is suitable for big data processing. FastMap is an approximation of MDS that uses less memory and delivers higher performance than Classical MDS. The results of FastMap are not as accurate as those of the Classical MDS. FastMap integrates more disruptions. For the calculation of the 2D coordinates is dispensed with a square distance matrix and instead uses a distance matrix representation. On request, the desired distances are calculated on-the-fly. In summary, FastMap is suitable if a projection of large amounts of data (more than 3000 objects) is performed. If a memory leak occurs on the Classical MDS, FastMap should be used. FastMap dispenses with a distance matrix and is faster and more memory efficient than classical MDS.

Sammon mapping is a special case of the Distance Scaling MDS method. The goal is to consider smaller distances with the same relevance as larger distances.

ICA (*Independent Component Analysis*), describes a method to determine independent components. The method seeks to determine statistically independent factors that do not interfere with each other. The method is recommended if a clear separation of the individual components is required. The process removes all similarities between factors.

In summary:

– PCA: linear relationships of data from a data foundation are projected onto 2D coordinates. Distance matrix optional.
– Kernel PCA: Non-linear relationships are projected onto 2D coordinates. Distance matrix optional.
– MDS: 2D coordinates are determined on the basis of a distance matrix. For metric scaled data, ClassicalScaling should be preferred to DistanceScaling.
– FastMap: Heuristic variant of the MDS, suitable for MDS projections of large amounts of data.
– Sammon Mapping: Special case of MDS, trying to avoid favoring larger distances over smaller distances.
– ICA: Determination of statistically independent components that have no commonality.

2.6   Visualization

The 2D coordinates from the projection are drawn on a display surface. The representation is similar to a scatterplot. The similarities between the different objects are detected faster with this visualization than with the tabular representation. Similar objects are close together, whereas dissimilar objects are far apart.

Zooming Concept. Due to the limited display area, a large amount of data points can not be visualized. With large data sets of objects, objects may overlap or be drawn on top of each other, so that unambiguous assignment is not

possible. One possible solution to prevent overplotting is to use a zooming concept. By using a zooming concept, the display area can be enlarged as required so that larger amounts of data can be visualized at the individual zoom levels. A semantic zoom concept allows you to visualize a large amount of data points without overplotting [12].

With a semantic zoom, the display area increases with increasing zoom level, so that more details, i.e. additional objects can be displayed without creating overlaps. Based on Pad [1], a zooming & panning concept is used. With zooming, the user enlarges the display area and with panning he navigates within the current zoom level on the display area. For the development of a suitable zooming approach, a compromise must be found between preventing overplotting and little loss of detail. An algorithm that does not represent objects, even though there is enough space on the canvas, would be inappropriate. The semantic zoom approach should use the display surface effectively and represent all data points if there is enough space. As the zoom level increases, the display area increases, leaving more room for more detail. Therefore, it is necessary that objects of lower zoom levels do not disappear, but are displayed and additional details appear [7,8]. Above all, it is important to rank the various objects in order to classify them as important or unimportant [6]. A suitable decision criterion for determining a ranking describes the measure of interest. Based on the measure of interest, the objects are sorted according to their importance in descending order. The measure of interest can be defined by the user by choosing an appropriate attribute. A metric scaled feature is selected and the feature's data is sorted in descending order of size. Objects of high importance are above the leaderboard, whereas unimportant objects are in the lower part of the leaderboard. The zooming processes take into account the ranking and distribute the objects at the different zoom levels. In this work, two zooming techniques have been developed, namely a bucketing approach and a neighbor approach.

Heatmap. The heatmap is a visualization technique to clarify relationships within the data. Heatmaps model the intensity or density between the data values. Depending on the intensity of the relationships, different colors are used. Often, warm colors are used for intense relationships (high density regions) and cold colors for weak relationships (low density regions). A heatmap makes it possible to visualize especially dense areas that create overplotting and interfere with a normal presentation. The heatmap is provided as an additional layer that can be hidden or hidden as needed. The variants static and dynamic heatmap exist for the representation of a heatmap. A static heatmap is displayed identically at all zoom levels, regardless of the zoom level, whereas a dynamic heatmap looks different at each zoom level. With a dynamic heatmap, a separate heatmap is created for each zoom level, which changes as the zoom level increases.

Presentation. This section describes concepts for the graphical design of the individual data points. In the visualization pipeline various techniques are described for generating a visual representation from a given data set. The graphical design of the individual data points is relevant to distinguish between individual objects. Black dots of equal size on a scatterplot are insufficient to

illustrate relevant details. Although relationships between the individual objects
are recognized, the objects do not visually differ from each other. The user
does not recognize individual properties of the respective objects. Using a suit-
able visualization technique, data points could be customized so that individual
properties of the objects are visible. Economic relationships between different
countries could be visualized with the help of flags or air quality through objects
of different colors. A red color describes a deficient air quality, whereas green
indicates a good air quality. Therefore, approaches are useful in which objects in
color, size, shape vary from each other and can be illustrated by different images.
For this purpose, three approaches have been defined to adapt the data points in
the simplest way. With a picture mapping every data point can be represented
by a picture. The user is allowed to use images from web pages or local images
from the computer. Another option is the Icon Wizard. By selecting different
fields, different parameters can be set to visually represent objects. OData intro-
duces an abstract modeling language that provides different configurations of
the objects. This language can be used to assign different shapes or images to
an object.

2.7   Optimizations

The model from sections generates a visual representation from a multidimen-
sional tabular file. A presentation and a semantic zoom optimize the presentation
of the data points. The basic problem is that the processing steps are complex
and therefore do not generate the required map material in real time. However,
it is required that the map material be provided to multiple users simultane-
ously in real time. Map Tiling is a concept in which the resulting maps are
stored on the hard drive. The map material is divided into sections and can be
loaded from the hard disk on request. The user would like to compare several
combinations consisting of different features as well as different configurations.
The concept of pre-calculated map sections is unsuitable for managing multiple
configurations because only one configuration per map set can be created. With
a large number of variations - created by using different tabular files and several
different configurations - a large number of card sets are created under different
names. Management using different names makes it difficult to understand large
volumes of card sets. The Lazy Evaluation Approach introduces a complex file
system for managing various feature combinations under a map set. As a result,
different feature combinations can be created under a card name.

3   Usage Scenarios

3.1   Iris Dataset

The iris dataset is a popular dataset for testing classification methods. In a clas-
sification, the choice of relevant features tries to determine a correct assignment
to the given classes. The dataset contains 50 objects with the explanatory vari-
ables sepal-width, sepal-length, petal-width, petal-length and the variable class

to be determined. The goal is to use the four explanatory variables to determine the class. The variable class describes the three types of flowers Iris Setosa, Iris Versicolor and Iris Virginia. The flowers are presented in Fig. 2. The three flowers have a sepal (= sepal) and a petal (= petal). Based on Fig. 2, the sizes sepal and petal are described. In the picture sepal describes the yellow leaf and petal the green leaf of the flower. From the sizes petal and sepal, the width (= width) and height (= height) are recorded in the dataset. The table in Fig. 1 shows the structure of the data and describes part of the data set.

sepallength in cm	sepalwidth in cm	petallength in cm	petalwidth in cm	classes
5.1	3.5	1.4	0.2	Iris-setosa
6.3	3.3	6.0	2.5	Iris-virginica
7.0	3.2	4.7	1.4	Iris-versicolor
...	...	...	...	...

Fig. 1. Iris data set

Fig. 2. Left: sepal and petal. Right: flower types Iris (Color figure online)

The following section describes in several steps a procedure for analyzing the data record using the visualization system. After the iris dataset has been loaded into the system, a selection of all explanatory variables - sepal-length, sepal-width, petal-length and petal-width - is made to explain a classification of the different flower types. In a first test, an Euclidean distance is selected with a FastMap projection, without a semantic zoom, and the objects are represented by black dots. From Fig. 2 it can be seen that there are relationships between the objects but no differentiation of the different flower types is possible.

A picture mapping based on the three flower types could help to clarify the connections between the objects. The image mapping of the iris dataset is shown in Fig. 3. A better differentiation of the different flowers has been achieved only partially compared to the black dots. Because of the same color (purple), the minimum size of the images on the display and the related form, the flowers can not be clearly separated. In addition to the image mapping, there is the possibility to
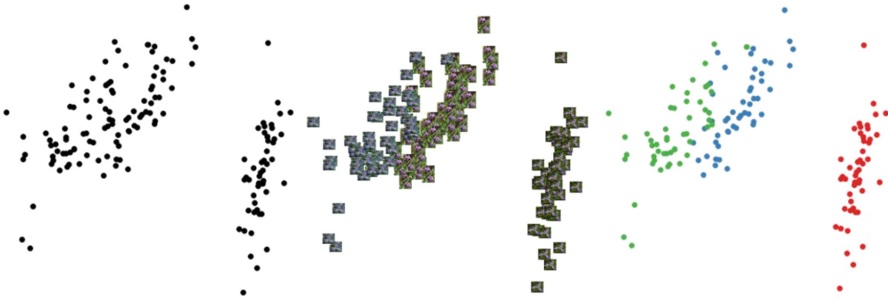
Fig. 3. From left to right: default scatterplot. ImagePlot IconPlot (Color figure online)

vary an object in its color and size. An icon mapping could help. The iris dataset describes a classification problem, so it makes sense to consider the classes as categories. A qualitative color map is suitable for better visualizing relationships between different categories. Figure 3 uses a Qualitative Colormap to visualize the different flower classes. With the different colors a clear separation of the classes is visible. The iris-virginica (= green) differs from the iris-versicolor (= blue) and the iris-setosa (= red). The separation becomes visible because the objects of the identical flower varieties are close to each other, whereas different varieties of flowers are far apart are. Nevertheless, it can be seen that the iris-virginica is more similar to the iris-versicolor than the iris-setosa. Using a sequential color map and varying the size, numerical properties of objects can be highlighted. High numeric values are represented by larger or darker objects. Whereas a lower value is represented by a lighter or smaller object. The width (= width) and length (= length) characteristics can be used to represent the size and color of the objects. A comparison of sepal length and sepal width (see Fig. 2) shows that there is a relationship between the two variables. A longer sepal (= sepal) tends to imply a wider sepal. Another connection can be seen in the petal. With increasing length of the petal (= petal) increases the width (Fig. 4).

The visualization system allows for exploration of relationships within the iris dataset, taking appropriate procedures into account. The visualization system makes it possible to display a data record in just a few steps. Taking appropriate procedures into account, optimal representations of the data are generated so that relationships within the iris dataset become visible. The Iris dataset has been transformed into a visual representation considering the four features - sepal width, sepal length, petal width and petal length. In no time, the data was displayed on a scatterplot with black dots. This representation is difficult for the user to interpret. Relationships between the individual objects were not recognized. The presentation of the data set could be suitably displayed by changing the presentation. With the help of the icon option objects are displayed in a simple and understandable way. By varying the color and size and taking into account inuitive color maps, the objects are individually adapted. Through a
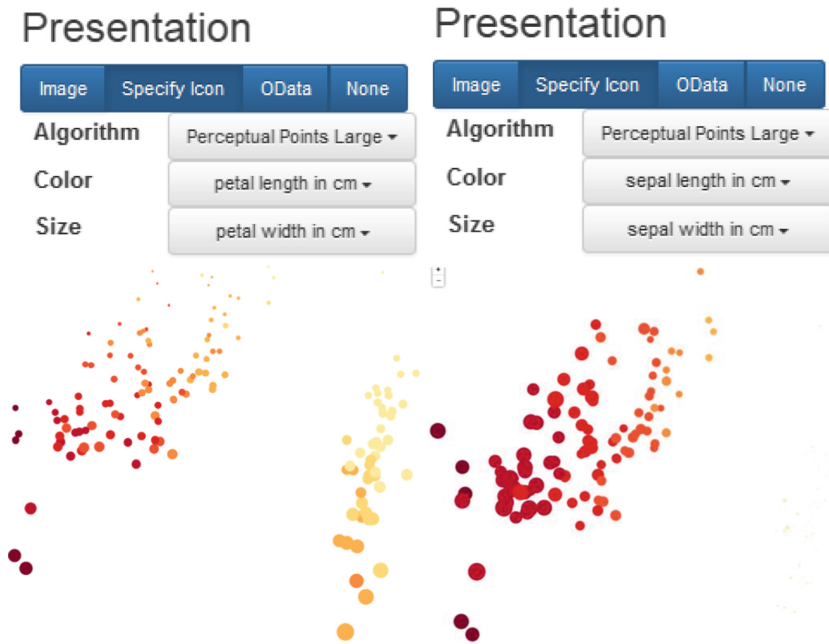
Fig. 4. Top: categorical and relationship presentation. Bottom: color sepal length & size: sepal width color - petal length & size: sepal width (Color figure online)

qualitative color map and constant size of the objects, connections have become visible. A clear separation of the different types of flowers became visible. The visualization system allows an analysis of relationships between two features of objects. Using a sequential colormap and the variation in size, relationships between the latitudes and lengths of sepal and petal have become apparent. A negative relationship between the width and length of the sepal, as well as a harmonious relationship between width and width of the petal were recognized.

3.2 OECD Dataset

The OECD dataset looks at the OECD Better Life Index The OECD Better Life Index makes it possible to compare well-being between different countries, considering eleven aspects. The eleven aspects are: "Common sense, education, environment, civic engagement, health, housing, income, employment, life satisfaction, security and work-life balance". The index describes the eleven aspects of the 34 OECD member countries, including Brazil and Russia. Every year the index is updated. Every year the data is collected. The Index provides citizens and scientists with information about people's quality of life and progress in society. The dataset (Fig. 5) contains 36 instances and is characterized by the characteristics of country, educational attainment, employee working hours (long hours), life expectancy, life satisfaction, health report (= self-assessment).

reported health), student skills, leisure time = time devoted to leisure and personal care, and school years = years in education.

Label	Educational attainment	Employees working very long hours	Life expectancy	Life satisfaction	Self-reported health	Student skills	Time devoted to leisure and personal care	Years in education
Brazil	43	10.74	73.4	7.2	69	402	14.97	16.3
Mexico	36	28.77	74.4	7.4	66	417	13.89	15.2
Chile	72	15.42	78.3	6.6	59	436	14.41	16.4

Fig. 5. Sample of OECD better life index data

The visualization system presented here is suitable for graphically depicting relationships. The objects are arranged using complex methods based on the similarities between the objects in space. To visualize the data, the record is uploaded from the local computer via the interface to the system. In the use case, all numerical entries (integer, double) are taken into account for the transformation of the data into a graphical representation. When choosing the none option for Semantic Zoom, no semantic zooming is initially considered so that all objects are displayed at each zoom level. The objects are initially displayed as black dots. For the calculation of the spatial coordinates a FastMap dimension reduction takes place, taking into account Euclidean distances. The selected configuration creates a scatterplot, as shown in Fig. 6. The similarities between the objects are recognizable, but a distinction between the objects is limited. On the black dot scatterplot in Fig. 6 countries can not be distinguished from each other. By means of a suitable visualization a differentiation is possible.
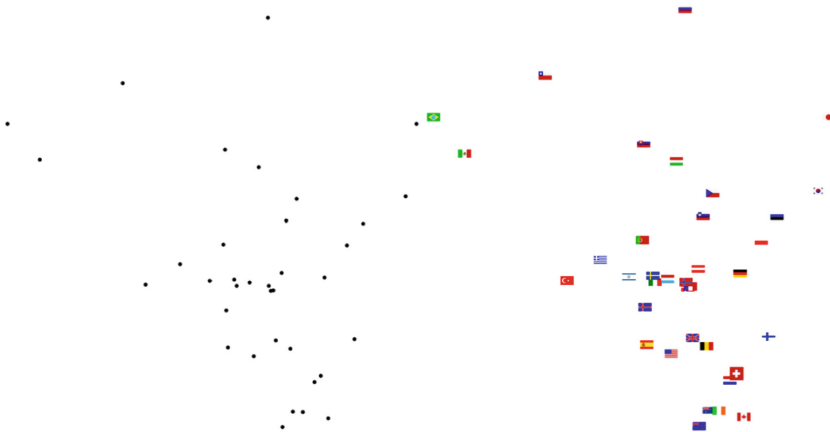

Fig. 6. Black dot scatterplot and image mapping

The OECD dataset compares countries with each other. A representation of the data points by their country flags allows the user to quickly capture relationships between countries. In Fig. 6, objects are visualized by their country flags. Compared to a black point scatter plot, interrelationships between countries are directly identifiable. The user sees at a glance which countries are related to each other. Figure 6 shows similarities between countries. Two countries are similar to each other when the distance is small. Countries between which a large distance exists are dissimilar and thus different from each other. There is a big gap between Russia and the rest of the country. In the distance calculation, all selected features are taken into account. Figure 7 compares Russia with other countries. The chart shows that the values of the characteristics Employees working very long hours, as well as Self-reported health are smaller compared to the other countries. The country Slovenia lists in the table compared to Russia a significantly higher value for the working time of the employees. The differences in size lead to large distances between Russia and the rest of the country. The countries Slovenia and Czech Republic are close to each other and are therefore similar to each other. The distance between the two countries is low because the values for life expectancy, self-reported health and student skills show little variation.
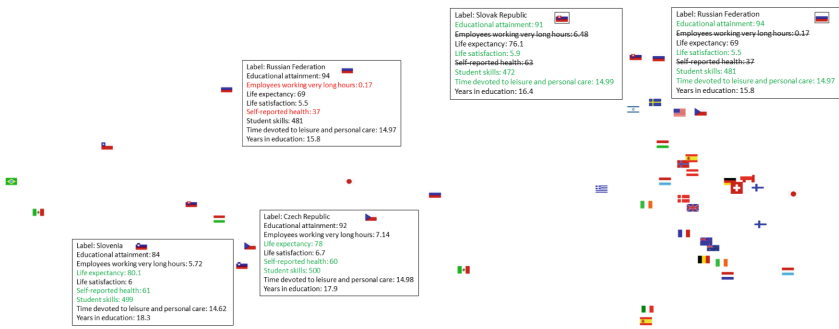

Fig. 7. Left: comparison of countries. Right: comparison between Russia and Slovakia

The distances between Russia and the rest of the countries exist because the characteristics of employees working very long hours and self-reported health deviate strongly. The visualization system allows a selection of different features so as not to consider them. For further consideration, the two features are removed from the further considerations. With the new feature combination a transformation is made with which a new representation of the ImagePlot is created. In the new illustration, Fig. 7, it can be seen that the distances between Russia and the other countries have narrowed. With the new feature combination Russia and the Slovak Republic are similar to each other. The gaps between the characteristics of "education attainment", "life satisfaction", "student skills" and "time devoted to leisure and personal care" are small, making Russia and Slovak Republic similar.

4   Conclusions

The aim of the work was to research and develop a web-based visualization system for creating and testing zoomable projection cards. The basic idea is to project a multidimensional dataset onto two dimensions using projection methods to represent it on a 2D scatterplot. For the development of the visualization system, several related papers were analyzed to identify the area of research and potentials. Required basic methods were elaborated to understand the peculiarities of the technologies used. Based on the methods and work, a concept has been developed to transform the data into visual representations. The implementation of the web-based visualization system was presented. Using various use cases, the advantages and potentials of the visualization system were presented using data examples.

References

1. Bederson, B.B., Hollan, J.D., Perlin, K., Meyer, J.M., Bacon, D., Furnas, G.W.: Pad++: a zoomable graphical sketchpad for exploring alternate interface physics. J. Vis. Lang. Comput. **7**, 3–32 (1996)
2. Boulos, M.N.K.: The use of interactive graphical maps for browsing medical/health internet information resources. Int. J. Health Geogr. **2**(1), 1 (2003)
3. Burkhardt, D., Nazemi, K., Breyer, M., Stab, C., Kuijper, A.: SemaZoom: semantics exploration by using a layer-based focus and context metaphor. In: Kurosu, M. (ed.) HCD 2011. LNCS, vol. 6776, pp. 491–499. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21753-1_55
4. Card, S.K., Mackinlay, J.D., Shneiderman, B. (eds.): Readings in Information Visualization: Using Vision to Think. Morgan Kaufmann Publishers Inc., San Francisco (1999)
5. Gutbell, R., Kuehnel, H., Kuijper, A.: Texturizing and refinement of 3D city models with mobile devices. In: Blanc-Talon, J., Penne, R., Philips, W., Popescu, D., Scheunders, P. (eds.) ACIVS 2017. LNCS, vol. 10617, pp. 313–324. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70353-4_27
6. Kuijper, A.: On detecting all saddle points in 2D images. Pattern Recogn. Lett. **25**(15), 1665–1672 (2004)
7. Kuijper, A.: Using catastrophe theory to derive trees from images. J. Math. Imaging Vis. **23**(3), 219–238 (2005)
8. Kuijper, A., Florack, L.: The relevance of non-generic events in scale space models. Int. J. Comput. Vis. **57**(1), 67–84 (2004)
9. von Landesberger, T., Fiebig, S., Bremm, S., Kuijper, A., Fellner, D.W.: Interaction taxonomy for tracking of user actions in visual analytics applications. In: Huang, W. (ed.) Handbook of Human Centric Visualization, pp. 653–670. Springer, New York (2014). https://doi.org/10.1007/978-1-4614-7485-2_26
10. Lins, L., Klosowski, J.T., Scheidegger, C.: Nanocubes for real-time exploration of spatiotemporal datasets. IEEE Trans. Vis. Comput. Graph. **19**(12), 2456–2465 (2013)
11. Maus, M.: Definition und visualisierung von zoombaren 2D-projektionen im web. Technical report, TU Darmstadt (2016)

12. Nazemi, K., Breyer, M., Forster, J., Burkhardt, D., Kuijper, A.: Interacting with semantics: a user-centered visualization adaptation based on semantics data. In: Human Interface and the Management of Information. Interacting with Information - Symposium on Human Interface 2011, Held as Part of HCI International 2011, Orlando, FL, USA, 9–14 July 2011, Proceedings, Part I, pp. 239–248 (2011)
13. Pekalska, E., Duin, R.P.W.: The Dissimilarity Representation for Pattern Recognition: Foundations and Applications. Machine Perception and Artificial Intelligence. World Scientific Publishing Co., Inc., River Edge (2005)
14. Stahnke, J., Dörk, M., Müller, B., Thom, A.: Probing projections: interaction techniques for interpreting arrangements and errors of dimensionality reductions. IEEE Trans. Vis. Comput. Graph. **22**(1), 629–638 (2016)
15. Zöllner, M., Jetter, H.C., Reiterer, H.: ZOIL: a design paradigm and software framework for Post-WIMP distributed user interfaces. In: Gallud, J., Tesoriero, R., Penichet, V. (eds.) Distributed User Interfaces, pp. 87–94. Springer, London (2011). https://doi.org/10.1007/978-1-4471-2271-5_10