Interactive Visual Analysis of Multidimensional Geometric Data

Igal Milman
National Research Nuclear University MEPhI (Moscow Engineering Physics Institute)
115409, Moscow, Russia
igalush@gmail.com

Victor V. Pilyugin
National Research Nuclear University MEPhI (Moscow Engineering Physics Institute)
115409, Moscow, Russia
VVPilyugin@mephi.ru

ABSTRACT

One of the most important tasks in modern world is to find solutions to problems of processing and analyzing multidimensional data. In this paper we present an approach for cluster analysis of multidimensional geometric data. Some definitions and extensions of classical cluster analysis problem is given. Our approach is based on the visualization method. Suggested approach allows us to analyze multidimensional data and distances in multidimensional Euclidean space using three-dimensional spatial scenes and shows an easy way for cluster analysis and anomaly discovery. An example of solving the problem of analysis of financial multidimensional data of credit organizations is also presented.

Keywords

Visual analysis, cluster analysis, multidimensional data analysis, visualization.

1. INTRODUCTION

One of the most important tasks in modern world is to find solutions to problems of processing and analyzing multidimensional data. Different methods and procedures, both automatic and interactive, have been developed to solve such problems. Visual methods take a special place among the data analysis problem solving methods.

However, a careful study of publication focused on the description of specific applications that use visual methods, allows us to state that in reality, interactive multidimensional data analysis systems often have a lower value than systems displaying results gotten using data analysis methods. As an example we can use situational alerts system AdAware [1], system of visual analysis system that is used to solve problems in aircraft manufacturing [2], system of visual analysis of text data VxInsight, software package SAS Visual Analytics [3], created to process and analyze large volumes of financial and economic data. All above mentioned systems are industrial and commercial products; they provide users with a great number of interfaces and data visualization capabilities. However, while all of these systems, in fact, are set to process multidimensional data internally and present results in the form that is convenient for the user, they don't give him an option to work directly with data clouds using multivariate visual display of the data.

As practice shows methods of parallel coordinates [4], Chernoff faces [5], Andrews plots [6] and other mnemonic graphic images are widely used for such visual representation for multidimensional data. Such images have a set of settings corresponding to the coordinates of multidimensional point. And, by comparing that images, one can cluster the initial data. For more info about such methods see works [4-6]. These methods do not allow the user to use any kind of metrics to understand the difference between objects. That is a crucial point, if the analyst has to answer the question "why does this objects are similar?"

This article discusses an original algorithm that we developed to solve problems of multi-dimensional geometric data analysis. Justified choice of the method used preceded by the development of the algorithm and based on the algorithm we created an interactive software package for visual analysis of multidimensional data. This method and algorithm are different from others since they provide the user with the ability to work directly with the original multidimensional data - there is no initial numeric processing of the original multidimensional data, and that allows analyst to manipulate directly with input data and visually analyze the results.

2. STATEMENTS OF THE GEOMETRIC DATA ANALYSIS PROBLEM

In this article, a geometric data refers to a set of points $(x_1, x_2, \dots, x_n)$ of Euclidean space E_n with predetermined metric tensor $\rho(x, y)$, which may be prepared by geometrization of any domain data. The task of geometry data analysis is understood as a problem of extended cluster analysis, as well as the imposition of additional statements on the mutual positions of the points in multidimensional geometrical space.

2.1. The classical problem of cluster analysis

The problem of cluster analysis is one of the classical problems of data analysis. Setup of goals of cluster analysis includes the following:

Given: set of points $G = \{x_1, x_2 \dots x_m\}$, where $x_i = (x_i^1, x_i^2, \dots, x_i^n)$

Required: divide subset G_i from G, in a way that:

1) $G_i \cap G_j = \emptyset, \forall I, j, i \neq j$
2) $\cup G_i = G$

In the classical statement of the cluster analysis problem, subset G_i is called *cluster* and must satisfy the following conditions (with certain function for calculating the distance $\rho(x, y)$ and maximum intra-cluster distance d):

1) $\forall x, y \in G_i, \rho(x, y) \leq d$
2) $\forall x \in G_i, \forall y \in G, y \notin G_i, \rho(x, y) > d$W

2.2. Extended problem of cluster analysis

Depending on the distance $\rho(x, y)$ between the points and the parameter d, that may be changed during the analysis process, it is possible to visually distinguish the following subset of multidimensional points:

1. *Cluster* — classical cluster.
2. *Remote (anomalous) point* — a point x_i is remote, if $\forall y \in G, \rho(x_i, y) > d$. We may say, that distant point — is a cluster of the size of one. However, these points may be of particular interest for the analyst.
3. *Bunch* — a subset of points with most distances between points not exceeding the preset d value.
4. *Quasi-remote point* — a point that is not remote, but at the same time is not included in a bunch or a cluster at the given grouping.

Note, that the analyst selects bunches and quasi-remote points during the process of solving the above-mentioned problem of the analysis. These concepts are useful for the analyst in the process of solving the problem of geometric data analysis.

Allocation of bunches and quasi-remote points allows the analyst to focus on these objects during the process of changing the parameter d. Therefore, if there is an allocated quasi-remote point, it is necessary to gradually change the value of d, to find out the conditions under which a point would become anomalous. Similarly, when allocating bunches, it is necessary to change d to try to obtain a cluster.

2.3. The statements of the relative positions

In the process of solving the problem, statements of the following types are made:

- Point x_i belongs to subset G_j when $d = d_k$
- Subset G_j is a cluster
- Subset G_j is a bunch
- Point x_i is an anomalous point
- Point x_i is a quasi-remote point.

As a result, in this publication we are solving the problem of partitioning of the original multidimensional geometric data into subsets, such as clusters and remote points, as well as the allocation of bunches and quasi-remote points supporting the problem solving process when the analyst changes maximum intra-cluster distance d.

3. THE PROPOSED METHOD

To solve this problem, it is proposed to use the visualization method. Theoretical aspects of the solution for data analysis problems with this method using the scientific data as an example are given in [7]. The essence of the visualization method is to divide the original problem into two consecutively solved sub problems. First problem, solved by a computer, is to obtain a representation of the analyzed data in a graphical display (the problem of data visualization). Second one, is to analyze the graphic image and interpret the results of the analysis against the original data. This problem is solved directly by man.

It is emphasized, that in this method the visual analysis of a graphical representation of the analyzed data is to qualitative analysis of the spatial scene that within this method is corresponding to the analyzed data. I.e. used graphics are means to naturally and comfortably for the analyst to visually analyze the spatial scene, followed by the interpretation of the results correlated to the original data. An algorithm for solving the first problem involves the following steps:

1. Sourcing — receiving original data for visualization pipeline.
2. Filtering — pre-processing the original data. During that step an interpolation of missing data, data decimation and data smoothing can be applied. In general, this step may be absent.
3. Mapping — on this step, the filtered data is mapped to spatial scene. This step is one of the most important and time-consuming in the first task.
4. Rendering — obtaining the resulting graphics of spatial scenes.

The second objective is to analyze the resulting graphics that is visual analysis of spatial scenes. This step cannot be strictly formalized, its effectiveness depends on the experience of the person performing the visual analysis and his tendency for spatially-shaped thinking. Looking at the resulting image, a person can solve 3 main objectives: analysis of the shape of spatial objects, analysis of their mutual disposition and analysis of graphic attributes of spatial objects. The results of the solutions of these three problems, as indicated above, are interpreted with respect to the original data.

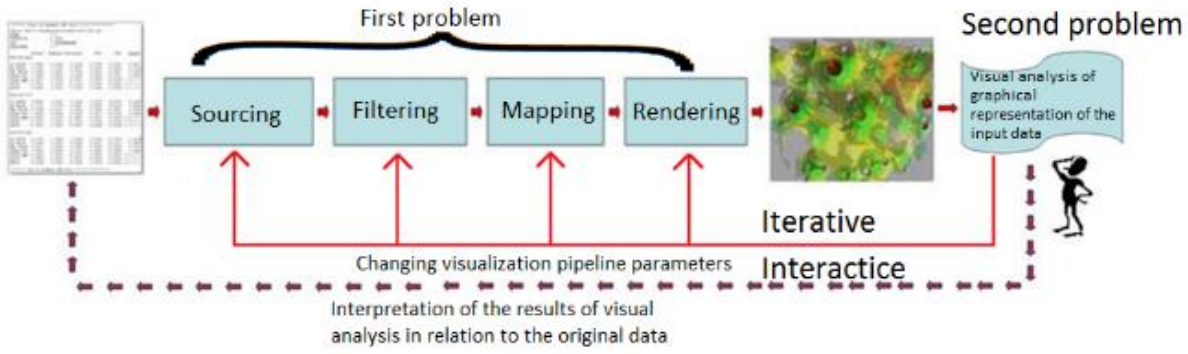


Figure 1. The general scheme of the visualization method

In the analysis of the graphics, the user can either be satisfied with the conclusions, or it may come back to one of the stages of the first task to set other values of the visualization pipeline parameters. Therefore, the process of solving the analysis' problem using the visualization method is iterative and interactive. The general scheme of analysis' problem solving is shown in figure 1.

3.1. Description of the basic idea of the algorithm

Under this method, we proposed an original algorithm for solving the problem of visualization. The basic idea is that in the original n-dimensional space E_n an additional construction is undertaken. If the distance between two n-dimensional points x_i, x_j is not more than pre-assigned d ($\rho(x_i, x_j) \leq d$), line segment is drawn between two points in the original space. Accordingly, the original analyzed data appeared to be m multidimensional points and some multidimensional line segments (depending on d). Then projection of the original space on the selected by the analyst 3-dimensional space X_i, X_j, X_k is performed. Next, a spatial scene is constructed using the following rules:

- points correspond to spheres with preassigned radius;
- line segments correspond to cylinders with preassigned radius.

The color of the spheres is set to be the same, and the color of the cylinders depends on the distance in the original space. The smaller the distance, the redder the cylinder between the spheres. When setting up the color of the cylinder in the RGB palette, the color will be set as follows:

$$RGB = [255, 0, 0] + [-255, 150, 255] * \frac{\rho(x, y)}{d}$$

Setting various colors to cylinders allows making statements about the distance in the original n-dimensional space while visual analysis is performed in the 3 dimensional spatial scene.

In case of several n-dimensional points are projected into one 3-dimensional point, we should move for a bit one of the 3-dimensional point in a such way, that the points are not overlaying anymore. Due to the fact that described algorithm assumes analysis of the distance between n-dimensional points, such transformation does not violate the process of visual analysis of the spatial scene and the analysis of the initial data as a whole. So, even in that case, that algorithm is valid.

3.2. Detailed description of the algorithm

The algorithm of solving the geometric data analysis problem is represented by the following steps:

1. Input of initial data.
2. Choosing the distance formula.
3. Setting the maximum intra-cluster distance d, calculating distance between every couple of points in the original n-dimensional space.
4. Entering visualization parameters (radius of the spheres and cylinders, space X_i, X_j, X_k for projection).
5. Projecting objects of the original n-dimensional space into chosen in step 4 3-dimensional space.
6. Creating of spatial scene.
7. Visualization and analysis of spatial scene.
8. If not all the necessary information is obtained, it is necessary to go back to step 3.
9. The results of the analysis were then interpreted relative to the original multidimensional geometric data.

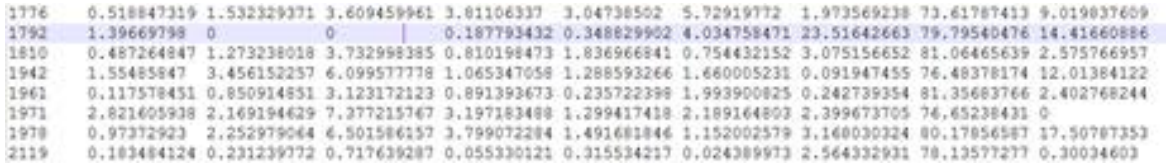
Therefore, the algorithm of solving the problem is an interactive and iterative.

Now we are estimating the complexity of one cycle of the algorithm (steps 3-8). If the number of points is defined as n, then the number of cylinders will be $n * \frac{n-1}{2}$. Thus,

$$T(n) = O\left(n * \frac{n-1}{2}\right) = O(n^2)$$

$$M(n) = O\left(n * \frac{n-1}{2} + n\right) = O(n^2)$$

Where $T(n)$ shows the dependence of the operating time on the volume of the input data (n);



1776	0.518847319	1.532329371	3.609459961	3.01106337	3.04738502	5.72919772	1.973569238	73.61787413	9.019837609
1792	1.39669798	0	0	0.187793432	0.348829902	4.034758471	23.51642663	79.79540476	14.41660886
1810	0.487264847	1.273238018	3.732998385	0.810198473	1.836966841	0.754432152	3.075156652	81.06465639	2.575766957
1942	1.55485847	3.456152257	6.099577778	1.065347058	1.288593266	1.660005231	0.091947455	76.48378174	12.01384122
1961	0.117578451	0.850914851	3.123172123	0.891393673	0.235722398	1.993900825	0.242739354	81.35683766	2.402768244
1971	2.821605938	2.169194629	7.277215767	3.197183488	1.299417418	2.189164803	2.399673705	76.65238431	0
1978	0.97372923	2.252979064	6.501586157	3.799072284	1.491681846	1.152002579	3.160030324	80.17856587	17.50707353
2119	0.183484124	0.231239772	0.717639287	0.055330121	0.315534217	0.024389973	2.564332931	78.13577277	0.30034603

Figure 2. Fragment of the original data

$M(n)$ shows the dependence of the consumed memory vs. the volume of input data (n).

3.3. Implementation in Maxscript and C ++ (VTK)

This algorithm has been implemented in two different ways. First it was implemented using the programming language maxscript (3ds Max environment) due to its simplicity and richness of conceptual apparatus and therefore high speed programming on it. Because of the high complexity of the resulting spatial scene and a large number of objects on it (with a 90-points in the scene is displayed up to 8000 objects), as well as high frequency of the redrawing of the spatial scene, rendering takes a long time (4-5 minutes), and a greater number of original points causes a memory overflow. An additional constraint imposed by the 3ds Max is the complexity to design the user interface and simplicity of the tools for its implementation.

Then, given these drawbacks of 3ds Max usage, it was decided to move to the C++ programming language using VTK 7.0 library for visualization and programming environment Visual Studio 2013. The program uses a total of 13 user-defined classes and 5 user-defined types.

Optimization of calculations, the usage of a compiled language instead of interpreted and simpler visualization software in the C++ version of the software helps streamline the rendering process. When we process data contained of 81 points, using the software, instead of 4-5 minutes before, it took a few seconds now. The amount of RAM required for such data in 3ds Max was close to 1GB, while the software written in C ++ requires only 70MB. As a result, the transition to a new language will allow to analyze much larger volumes of data, as well as to create user friendly interfaces and tools for manipulation of spatial scenes.

4. Example of using the software

The described software tool has been tried to solve the problem of data analysis on the activities of credit organizations, presented in tabular form. [8]

4.1. Characteristics of the original data

Original data is multidimensional tabular data obtained from the financial statements of 81 credit organizations with 9 parameters for the second half of 2013 and the first half of 2014. A separate table was created for each month.

The tables have been created as follows: rows contain information about credit organizations, and columns contain parameters of those organizations. A total of 13 months was considered, so there are 13 tables. A fragment of the original data is shown in figure 2.

We tried to solve the problem of analysis of similarity of credit organizations. The goal was to highlight anomalous objects at different values of the similarity measures.

It was necessary to allocate credit organization diverged from other ones.

To solve the above problem using the proposed method we perform a geometrization of the problem. Geometrization allows us to transfer initial data from any domain to geometric data. Thus, after the geometrization, we can use described algorithm for any kind of data.

Geometrization will be performed as follows:

1. Each credit organization (each row of the table) will be assigned to a point of 9-dimensional Euclidean space.
2. credit organizations parameters (columns) will be interpreted as coordinates of points in the 9-dimensional space.
3. The distance in Euclidean space will be interpreted as a measure of the difference between credit organization. In this problem we use the Euclid distance:

$$\rho(x,y) = \sqrt[2]{\sum_{i=1}^{9}(x_i - y_i)^2}$$

4.2. Analysis

The algorithm of usage of the software requires to set a large value of the maximum intra-cluster distance. In other words, select a value d, in which all spheres are connected by the cylinders.

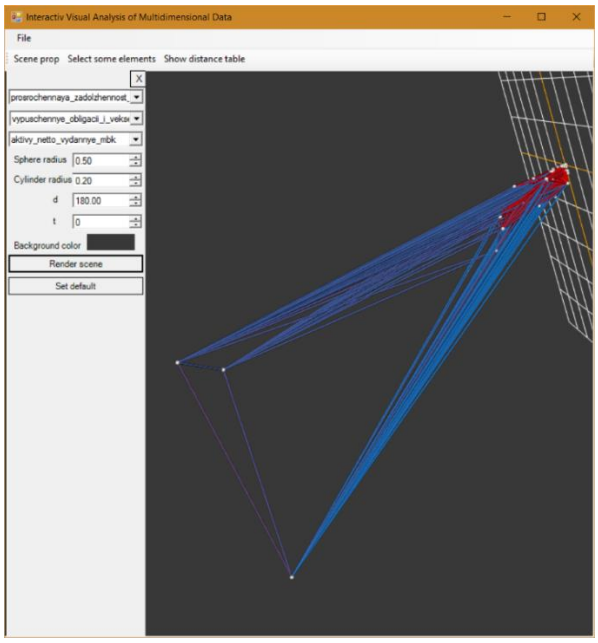


Figure 3. A graphical projection of the space scene, if *d=180*.

Figure 3 demonstrates a graphical projection image of the space scene, if *d=180*. As it is seen, all spheres are linked to each other and, therefore, respective multidimensional points made up a cluster. This value of *d* will be used as the initial value and it has to be reduced later on.

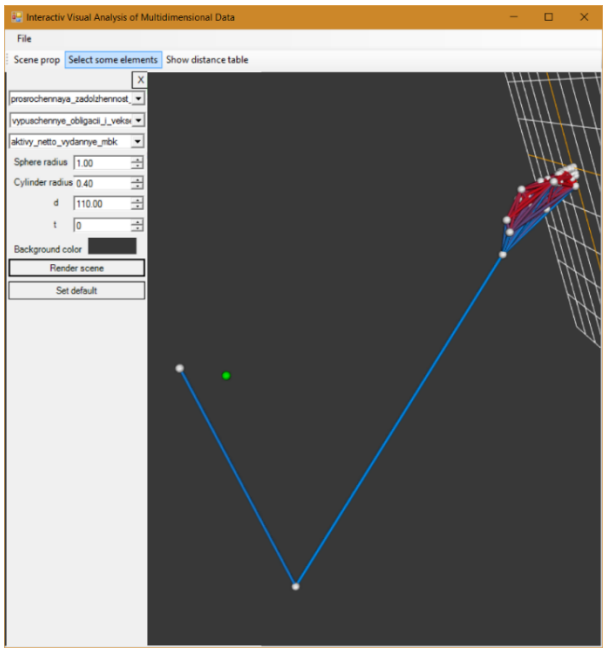


Figure 4. A graphical projection of the space scene, if d=110

Figure 4 illustrates a graphical projection image of the space scene, if d=110. We marked the sphere that has no connections with at that value of *d* as well as the appropriate remote point (ID=2748) by green color. Later on the color of the sphere will define the point in the multidimensional space fixed by the analyst, i.e. a predetermined color will allow us to trace any given point. With a further decrease of the d, highlighted green point will not change its properties, and there is no further need for its consideration. A bunch, containing of two white spheres, can be highlighted with this value of parameter d. Perhaps with further decreasing of the d, it will be turned into a cluster or two remote points.

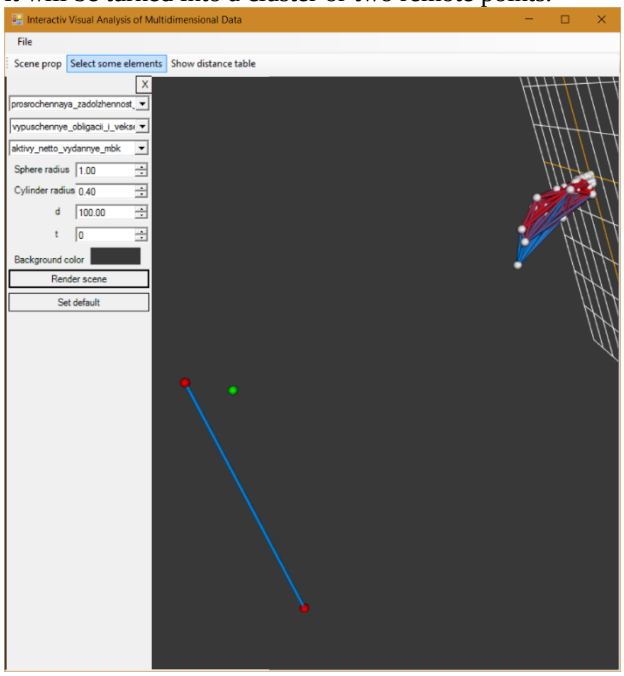


Figure 5. A graphical projection of the space scene, if d=100

Figure 5 represents a projection image of the space scene, if *d=100*. One can see that two spheres have been disconnected from others and two corresponding multidimensional points (ID=354 and 1000) formed a cluster. Based on the color of the cylinder being close to bright blue, the distance between these points is close to d. We will color these spheres (and the corresponding multidimensional points) in red.

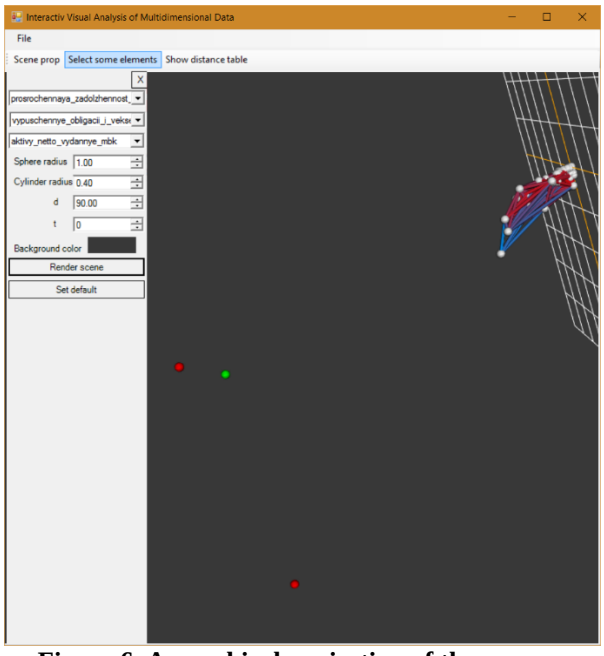


Figure 6. A graphical projection of the space scene, if *d=90*

Figure 6 demonstrates a graphical projection image of the space scene, if *d=90*. The cylindrical linkage between them is gone, which means the distance between the corresponding points greater than 90. In this case, the points turned into remote points. These points will not affect the further analysis.

Therefore, an analyst can choose green and two red points as desired remote points or he can continue the analysis by implementing further decrease of d and finding new remote points according to his knowledge of the specifics of the analysis of credit organization problem being solved [8]. In other words, analyst can conclude that there are three remote points. Moreover, in the process of solving the problem, with a sequential decreasing of the d, analyst may form the following additional conclusions:

1. When $d \leq 110$, point with ID=2748 is anomalous.
2. When $90 < d \leq 100$, points with ID=354 and ID=1000 form cluster of size two.
3. When $d \leq 90$, point with ID=354 is anomalous.
4. When $d \leq 90$, point with ID=1000 is anomalous.

5. Conclusion

In this paper we described the original algorithm for solving the problem of the analysis of multidimensional geometric data. This algorithm, in disparity to other algorithms that are using the visualization method, offers the user the ability to work directly with the original multidimensional data using visualized projection of that data in three dimensional space. The original numerical processing of multidimensional source data is not performed; instead, the analyst directly manipulates the source data and then performs visual analysis of the resulting data.

This algorithm was implemented using the programming language C ++. The resulting software tool has been tested on the data on the activities of credit organizations. As a further development of the system, it is proposed to add a number of tools for viewing spatial scene with different values of the maximum inter-cluster distances.

REFERENCES

[1] Y. Livnat, J. Agutter, S. Moon, and S. Foresti, 2005. *Visual correlation for situational awareness*. In IEEE Symposium on Information Visualization, pp. 95-102.

[2] D.N. Mavris, O.J. Pinon, D. Fullmer Jr, 2010. *Systems design and modeling: A visual analytics approach*. 27th Congress of International Council of the Aeronautical Sciences ICAS.

[3] *SAS the power to know*, [Online]. Available: http://www.sas.com/en_us/home.html. [Accessed 26 1 2016].

[4] A. Inselberg. *Multi-dimensional graphics: algorithms and applications*. EUROGRAPHICS'86, North-Holland (1986) pp. 7-18.

[5] David L. Huff, Vijay Mahajan and William C. Black. *Facial Representation of Multivariate Data*. The Journal of Marketing, Vol. 45, No. 4 (1981), pp. 53–59.

[6] Andrews D.F. *Plots of High-Dimensional Data*. Biometrics. Vol. 28, No. 1, Special Multivariate Issue (Mar., 1972), pp. 125-136

[7] A. Pasko,V. Adzhiev, E. Malikova, V. Pilyugin, 2013. *Some Theoretical Issues of Scientific Visualization as a Method of Data Analysis*. the Lecture Notes in Computer Science series.

[8] I.E. Milman, A.P. Pakhomov, V.V. Pilyugin, E.E. Pisarchik, A.A. Stepanov, Yu.M. Beketnova, A.S. Denisenko, Ya.A. Fomin, 2015. *Data analysis of credit organizations by means of interactive visual analysis of multidimensional data*. Scientific Visualization. Vol. 7. No. 1. Pp. 45 - 64.

[9] J. Thomas and K. Cook, 2005. *Illuminating the Path: Research and Development Agenda for Visual Analytics*. IEEE-Press.

[10] D.A. Keim, F. Mansmann, J. Schneidewind, J. Thomas, and H. Ziegler, 2008. *Visual Analytics: Scope and Challenges*. Lecture Notes in Computer Science, pp. 76-90.

[11] J. J. v. Wijk, 2005. *The value of visualization*. IEEE Visualization. Pp. 79-86.

[12] *What is Visual Analytics?*, [Online]. Available: http://www.visual-analytics.eu/faq/.

[13] D. Keim, G. Andrienko, J.D. Fekete, G. Carsten, J. Kohlhammer, 2008. *Visual Analytics: Definition, Process and Challenges*. Information Visualization - Human-Centered Issues and Perspectives, pp. 154-175.