© Proceedings of IADIS International Conference *e-Society 2003*
edited by António Palma dos Reis and Pedro Isaías
June 3-6, 2003, Lisbon, Portugal
IADIS, pp. 365-372

ADAPTIVE 3D INTERFACES FOR SEARCH RESULT VISUALIZATION

Wojciech Wiza, Krzysztof Walczak, Wojciech Cellary
Department of Information Technologies, The Poznan University of Economics
Mansfelda 4, 60-854 Poznan, Poland
{wiza,walczak,cellary}@kti.ae.poznan.pl

ABSTRACT

A new approach to three-dimensional interactive visualization of data retrieved from web search engines as a result of execution of a user query is presented. In the proposed method, the visualization interface applied to a particular result is selected from a number of available interfaces depending on the search result properties and information provided by a user. Two types of interfaces are described in the paper: holistic interfaces for presenting data classified according to different properties of the result and analytical interfaces for presentation of detailed result information. The use of a particular interface depends on its readability in the context of particular search result. The general readability rules for 3D interfaces are also presented. Possible mappings of search result properties to different visual properties of the interfaces are discussed and examples of different visualization interfaces are presented.

1. INTRODUCTION

The World Wide Web consisting of billions of web pages may provide a user with almost every type of information needed. All kinds of content for education, research, entertainment and leisure can be found on the web. The main problem is to locate this bit of useful information in enormous volumes of not structured and not categorized data. To solve this problem search engines were created. However, as the amount of information gathered in the search engine databases grows, it becomes increasingly difficult to present them to the users in an understandable and manageable form.

Most of search engines offer only simple textual interfaces that do not permit users to fully exploit the search results. A typical search engine displays search results by dozens of documents in a page-by-page principle. Retrieval of the next part of the result requires user interaction. Since the retrieved data is presented in small chunks, the user cannot see a global picture of the result. It is not possible to group or categorize the presented data. The only possible user interaction with the search result is selection of one of the presented links. Most of the existing search engines also limit the maximum number of records presented to a user. Typically, after presentation of several hundred records, a user has to re-specify the query. This approach is imposed by a requirement to keep the response times minimal, but it also often restrains the user from accessing important information.

In the classical approach, documents constituting a search result are ordered arbitrarily based on a ranking algorithm specific to the particular search engine. Usually, the ranking algorithms used to calculate the relevancy factors of documents are sophisticated and take into account multiple different aspects of the user query and properties of the indexed documents. But even the most complicated algorithms can be decoyed by a specific design of the web page, e.g. by repeating keywords in pseudo-phrases not appearing in the page contents or by the use of popular keywords which are not semantically connected to the page. As a result addresses that do not really match the specified query may be presented as the most relevant. Furthermore, since users cannot change the ranking algorithm they do not have influence on the final order of the presented documents.

At the current stage, progress in presenting the search results to users requires switching from classical textual to more advanced graphical user interfaces. There were several attempts to create graphical interfaces for search systems. In most of them 2D graphics has been used, but there were also attempts to apply 3D visualization. Several projects were carried out in HCIL [6] where the problem of visualization of big volumes of information has been addressed [5][17]. Other examples of projects applying 2D visualization are Antarctica [1] and InXight [9]. The projects exploiting 3D visualization of search results resulted in development of several visualization methods like 3D cards augmented by visualization of semantic relationships between documents [10], city-like landscapes [1][14], or positioning of objects in the 3D space like in VR-VIBE project [2][18], or Cat-a-Cone project [3]. In the NIRVE project [4][11][15] the 3D visualization is enriched by a concept of data clustering. An example of a 3D visualization system accessible for web users is the ViOS system [16].

Most of the search engines with 3D visualization interfaces developed up to now have not reached technological acceptance and commercial use, mainly because of the following drawbacks:

- The applied 3D graphical interfaces visualized information in a single 3D environment; as a result, different volumes of information had to be presented in the same scene. In many cases this approach resulted in improper presentation of information and thus decreased user perception;
- A user was presented with a 3D environment, where each document was represented by a 3D object. In the discussed solutions there were no attempts to present aggregated information first, and then – in response to user interaction – more specific information;
- The proposed systems either allowed full user interaction with the interface but required installation of some dedicated software, or were based on open Internet standards (e.g., VRML) but lacked full interaction capabilities as a result of the shortcomings of the general purpose standards.

In this paper a new method of interactive adaptive 3D visualization (AVE) of search results returned by indexing search engines is proposed. The visualization interface applied to a particular search result is automatically selected from a number of available interfaces depending on the search result properties and information provided by a user. Different types of interfaces are available, e.g. categorizing interfaces for presentation of aggregated data, detailed interfaces presenting details of the documents found, and comparative interfaces for comparison of different search results.

The reminder of this paper is organized as follows. In Section 2 the concept of adaptive user interfaces is described. In Section 3 examples of different types of interfaces are presented. In Section 4 discussion over application of interface visual properties is presented. Section 5 summarizes the paper.

2. THE CONCEPT OF ADAPTIVE USER INTERFACE

The amount of information returned by an indexing search engine may vary significantly. As the response to a user query, the search engine may return several records or several hundreds of thousands of records. Consequently, it is not possible to create a single 3D environment capable to visualize the entire spectrum of possible search result volumes. In the AVE method, the visualization system selects from a number of available visualization interfaces the one that best describes the search result. Assignment of the appropriate interface is based on the search result quantitative and qualitative properties (see Figure 1) to maximize its readability. This process may be fully automatic, with the visualization engine using pre-programmed logic to select the best interface, or user-aided with a user selecting a set of preferred interfaces. The visualization engine may also present a user with a set of interfaces that fit the particular search result and a user may choose the best one in his/her opinion.

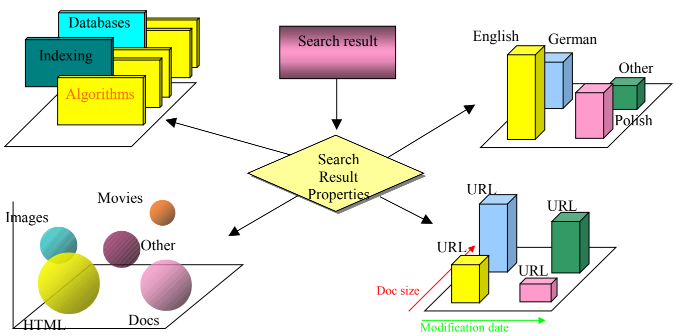


Figure 1. Selection of the visualization interface

The process of searching for the document of interest may be seen as a number of subsequent user queries where a query may narrow or broaden the previous search result. Using different levels of abstraction and applying the most appropriate 3D environment on each step, the AVE method permits to navigate from a high-level categorized, aggregated view of the entire search result, through categorized views of sub-results, up to precise visualization of information about particular documents of user interest. A user may formulate queries by adding and removing keywords, but also by interaction with 3D elements of the visualization environment (e.g., selecting objects by moving them to a predefined area). This multi-step process may be seen as a path through the visualized search results and is called *exploration path* (c.f. Figure 2). On this path the user is supported by an interface selection logic that helps to select appropriate interfaces. In sample exploration path presented, a user querying the search system with an 'Initial' interface, is presented with 'Spheres' interface, which groups results by domain. However, if all documents found are located on the same host, a 'Spheres' interface may be omitted and the result is visualized in 'Hedgehog' interface. Then a user may wish to visualize this same result using 'City' interface, and finally, browses documents of interest. Using specialized interfaces a user may also preview images and/or video files.

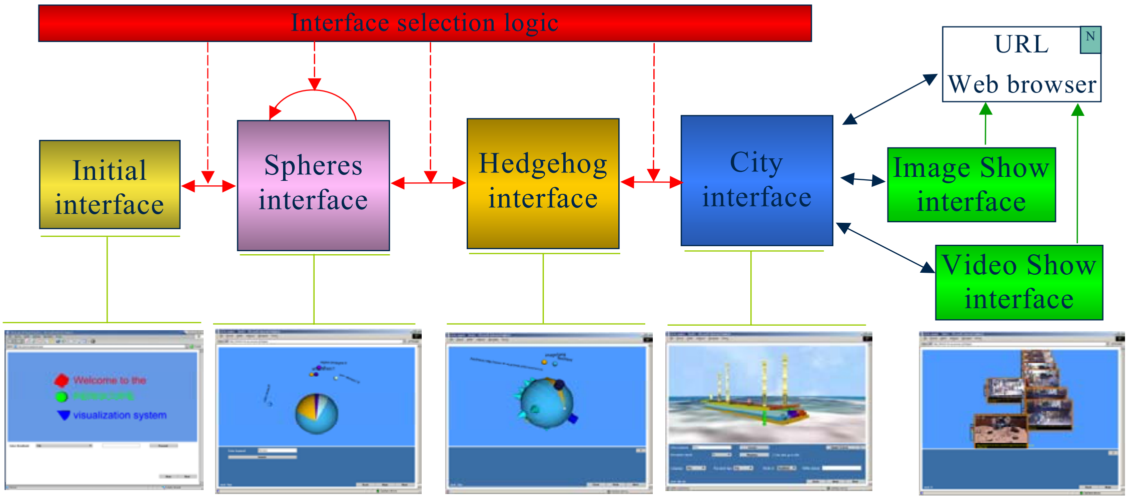


Figure 2. Sample exploration path

On each step of the AVE exploration path, the search result properties are mapped into one or more visual properties of a 3D scene. Each visual property, which may represent a single search result attribute, is called a *visualization dimension*. Visualization dimension may be represented as a property of an object (e.g., its color, size, shape, etc), or in another way, e.g., as a text associated with text node. Such assignment of document attributes to visualization dimensions may be dynamically changed by a user.

Selection of the appropriate interface, either automatic or manual, should produce a visualization that is readable to a user. While many different aspects of the readability may be discussed, some of them seem to be the most important in context of 3D environments. We assume that a visualization interface is readable if, and only if the following postulates are satisfied: (1) there exists a *viewpoint* from which all presented objects can be observed; (2) *size* of glyphs (where glyph can be defined as a single graphical object representing multivariate data object [20]) should allow their easy manipulation and interaction; (3) *occlusion* of each two glyphs in the scene should permit a user to interact with every glyph in the scene; (4) each interface dimension should represent a *unique* search result attribute (e.g., color represents document content-type); (5) the domain of each search result property is *properly transformed* into domain of a visualization dimension; and (6) *distance* between subsequent dimension values must be distinguishable by a user.

While both types of visualization – detailed and classifying – may be implemented by a single interface by changing only the meaning of visualization dimensions, in many cases a better visualization may be obtained by the use of specialized interfaces. Interfaces that are designed especially for presentation of detailed data are called *analytical interfaces* while interfaces designed to visualize aggregated data in a categorized way are called *holistic interfaces* (c.f. Figure 3).

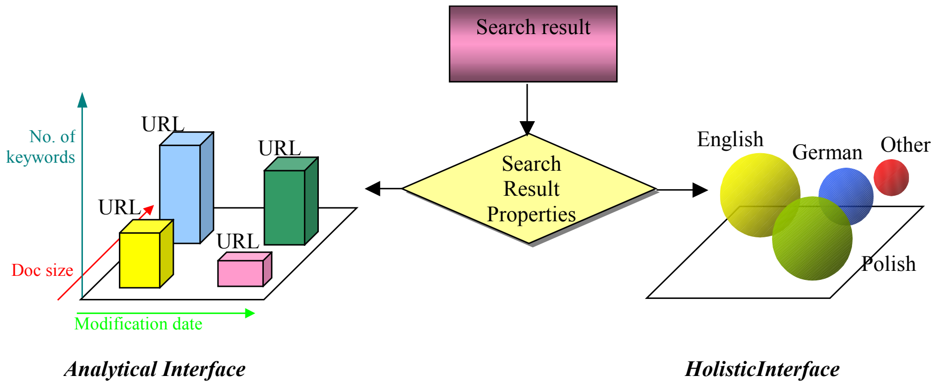


Figure 3. Selection of the visualization interface

An *analytical interface* is characterized by a high number of visualization dimensions (comparable with the number of search result attributes) and their visual separation. This permits detailed visualization of the search result allowing a user to observe and evaluate many different aspects of the displayed information.

Unlike an analytical interface, a *holistic interface* is characterized by a small number of visualization dimensions. Such interface can be used to present a generalized view of the search result where a user can instantaneously recognize the nature of the data but not the particular details. In a holistic interface, the categorization criteria are selected automatically based on the search result properties such as number of domains and/or sub-domains, number of sites, number of languages, document content-types, semantic relationships between documents, etc.

In both, analytical and holistic interfaces, the search result is always presented entirely. This permits a user not only to browse through the information but also to understand its nature. Appropriately constructed and applied interface permits a user to perceive trends in data faster and with bigger precision through clearly visible differences in colors, shapes, connectedness, continuity, symmetry, etc. A user may also apply different interfaces to the same search result in order to recognize different aspects of the same data set.

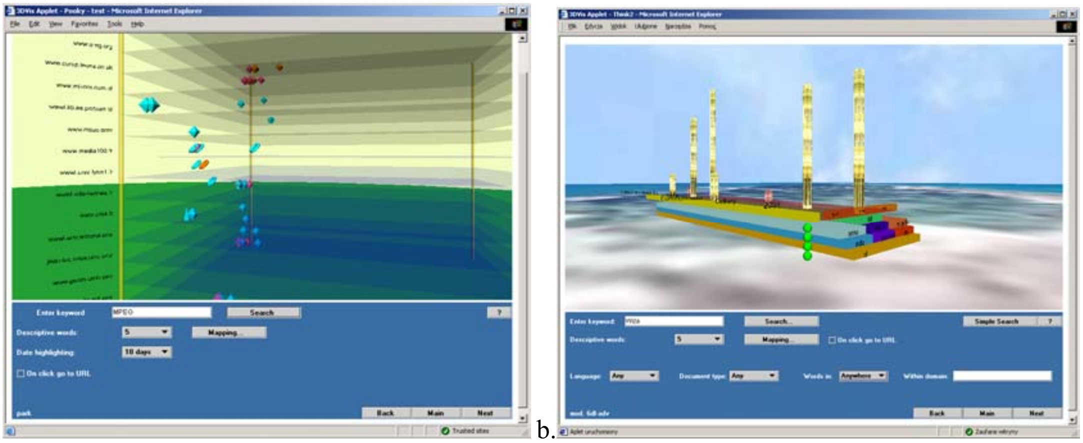
3. EXAMPLES OF VISUALIZATION INTERFACES

3.1 Analytical interfaces

Analytical interfaces are designed to present a user with detailed view of the search result. For this reason such interface should offer a high number of visualization dimensions. Typically, each object in analytical interface represents a single document, while the object properties reflect properties of the document. Therefore, with regards to readability prerequisites, such interface should be applied only to search results with relatively small number of records where visualization of document properties is also important.

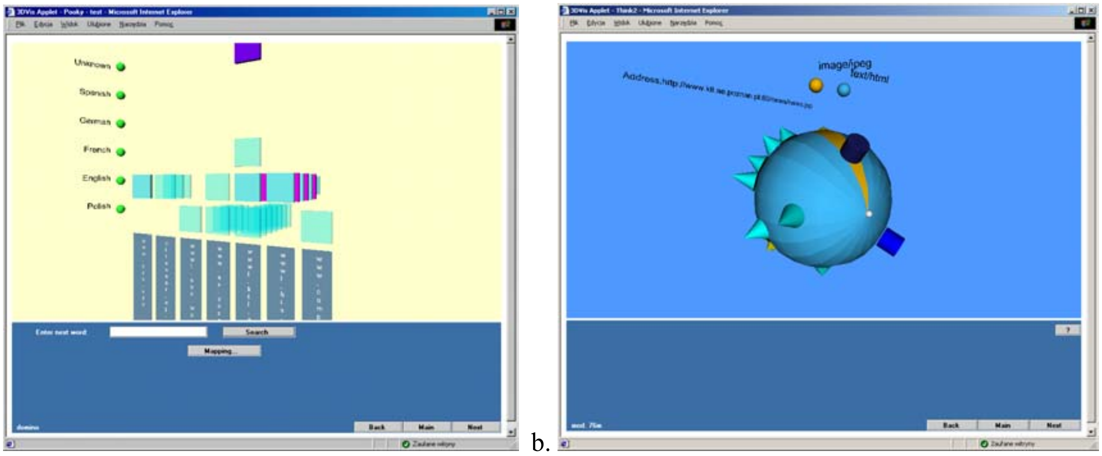
In Figure 4, two examples of analytical interfaces are presented. The interface presented in Figure 4a has 5 dimensions: shape representing language, color representing document type, position on Y-axis representing hostname, position on Z-axis representing document size, and position on X-axis representing document modification date. The interface presented in Figure 4b has 7 dimensions represented by X-, Y- (height of the floor) and Z-axis, color, texture, height and movement of the object.

In Figure 5a, an example of an incremental analytical interface is presented. With the use of this interface, a user can successively specify keywords in subsequent queries. After each query, tiles representing documents containing higher number the specified keywords become more red, while tiles characterizing documents with lower number of keywords fade into light blue color.



a. b.

Figure 4. Analytical interfaces with (a) 5 and (b) 7 dimensions



a. b.

Figure 5. Specialized analytical interfaces: (a) with incremental selection and (b) showing links on one web server

The analytical interface presented in Figure 5b can be used to present documents of interest existing on the same host. In this interface, object shapes denote document type, their color represents language, while position on the sphere corresponds to the number of keywords found in a particular document.

3.2 Holistic interfaces

Holistic interfaces are used to present a classified view of the search result. A holistic interface may show a search result classified using either one or several criteria, thus it does not need many visualization dimensions. Such interface may also contain some analytical elements permitting better evaluation of the search result. In Figure 6a, an example of a holistic interface with one classification criterion is presented. In this example, a sphere is divided into multicolored slices representing different Internet domains. Size of a slice represents the number of hosts containing documents of interest, hence a user may instantly recognize domains, where the probability of finding useful information is the highest. To improve readability of the interface, a sphere is surrounded by small colored bullets with textual tags providing names of the domains represented by particular colors.

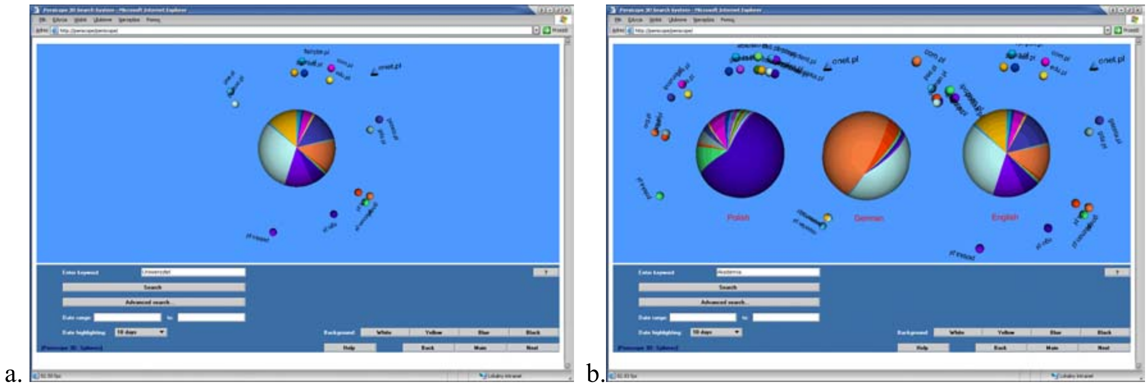


Figure 6. Holistic interfaces: (a) single criterion and (b) multiple criteria

An example of a holistic interface with two classification criteria is presented in Figure 6b. In this interface each sphere represents different languages of documents, while sphere segmentation represents Internet sub-domains.

In the Figure 7, a holistic interface which permits to compare two queries is presented. The interface is equipped with two input fields that permit a user to enter two different queries and compare their results represented as two series of coaxial cylinders. Each of cylinders symbolize one sub-domain containing hosts with documents of interests. Tiles attached to each of the cylinders represent documents. All documents residing on this same web server share the same color.

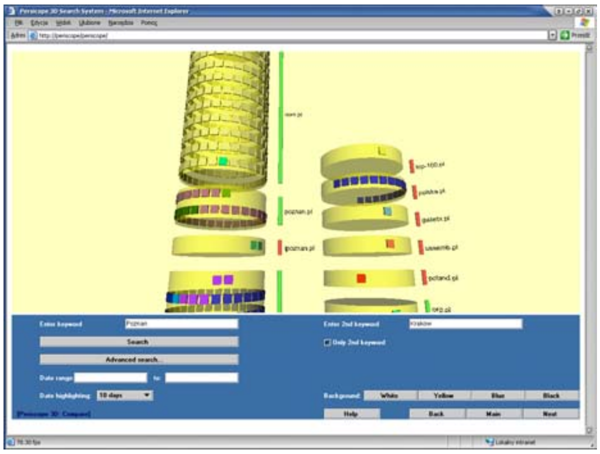


Figure 7. A holistic interface with analytical elements

4. USING INTERFACE VISUAL PROPERTIES FOR DATA VISUALIZATION

Creation of a readable interface that meets all prerequisites described in Section 2 requires understanding of the limitations of particular types of visualization dimensions. Each glyph can carry meaningful information through its position, size, shape, orientation, color and texture, and animation of each of the above.

Abilities to recognize differences in size and location are highly correlated. For example, if objects have dimensions of several meters, the difference in size or location measured in millimeters is unnoticeable. Although a size of a glyph consists of 3 dimensions, it is perceived by a user holistically. Assignment of different search result attributes to single geometric dimensions (x, y, and z) is possible, but is incomprehensible for a user and most likely would not be noticed.

Shape, with its potential variety, is a very capacious carrier of information. A high number of shapes can be recognized and distinguished by a user. The set of shapes that can be used for visualization consists of several geometrical primitives like sphere, cone, or cylinder; some well-recognizable polyhedrons like cube, tetrahedron or octahedron; and a big number of real-life shapes such as cars, books, furniture, trademarks, symbols, etc. The use of polyhedrons with growing number of vertices is limited because differences become difficult to distinguish (e.g. difference between dodecahedron and icosahedron).

Orientation of the glyph may be used only if the shape of the glyph permits to recognize the differences in orientation. Rotation of a cube may be in most cases easily noticed by a user, while the same transformation applied to a sphere cannot be identified. Small changes in orientation of a single glyph may be unnoticeable, but when this glyph is surrounded by a group of identical objects, even very small orientation differences may be immediately observed. Another problem is the change of orientation by certain angle along the glyph symmetry axis, which may lead to an impression, that no change was made at all (e.g. cube rotated by 90 degrees along an axis of symmetry).

Color of a glyph is a very flexible dimension. It has been estimated in [8] that a color monitor has between 2 and 6 millions of different colors available, but it is evident, that only a small subset of them can be perceived by a non-trained user. Other works [13] show that only a small number of colors can be used effectively as labels for expressing data. It is estimated that only 5-10 colors may be instantly recognized [7]. Moreover, recognition of color differences is highly related with luminance, hue value, contrast, saturation, monitor properties, and even human eye properties. A surface of the glyph may bring a large amount of recognizable information. A user may easily differentiate between plain color surfaces and textured surfaces. A number of distinguishable textures is nearly unlimited, while they vary in color and pattern. Even if two objects share the same pattern and color, texture orientation may be different (e.g., rotated by 45 degrees) bringing information to a user. This, however, applies only to well recognizable geometric patterns like parallel lines, squares etc. Rotation of texture imitating, for instance, stone surface, is unrecognizable.

Application of temporal dependencies to properties described above introduces additional informational dimension. Temporal coding is, however, limited in range and discreetness. The upper range limit depends on the property being animated: for some properties frequency of changes is limited by a visual inertia of a human eye and/or inertia of a display. For instance, a glyph changing its colors too fast is seen as having one. An object changing rapidly its position may not be visible on slow LCD screens. The lower limit of temporal coding is connected with: a) time when a user focuses on particular object (typically several seconds); and b) life time of the interface. An object property cannot be changed too slowly, because a user would not notice modification or the interface will be destroyed before.

The number of distinguishable levels in temporal coding is very low. A user may perceive a small difference in temporal changes of a particular property only if it can be compared to the original speed/frequency or another object. For instance, speed change of an animated object either has to be changed significantly or another animated object must be visible to permit speed comparison. The overall number of objects having time-dependent property values should be also kept small. In fact, temporal changes as an information medium may be used only sporadically. The interface, where a number of objects change their position, color, and shape in the same time is very likely to be unreadable.

The above issues should be carefully reconsidered while building interfaces, which may be used or designed especially for disabled people. In such case some properties cannot be used (e.g., color, when by people with inability to discriminate colors) or should be used with limited possible values (e.g. large differences in size of objects, in interfaces for people with partial blindness).

5. CONCLUSIONS

For testing and evaluation of the proposed AVE method, a prototype visualization system called *Periscope* has been developed [12]. The *Periscope* system is an intermediary system between users looking for documents on the Web and indexing search engines. The *Periscope* uses a set of interface models written in X-VRML language [19][21]. Interface models supplied with retrieved search result are presented to a user within standard VRML browser plug-in, like ParallelGraphics Cortona. Interface selection logic is also written in X-VRML.

First trials of the *Periscope* system connected to a custom search engine database containing information about 70% of sites within the Polish domain (.pl) proved that the AVE method can be efficiently used for Web searching. Although, the system response time is usually higher than in case of popular search engines (like Altavista or Google) reaching up to 30 seconds for complex queries, the end-users felt that the accuracy of the information retrieved was higher, especially in the cases when the initial constraints were not quite precise. During the tests it turned out that users often use several different visualization interfaces with the same query for better localization of the desired data.

Current tests focus on the ergonomics and perception of the 3D interfaces. Another group of tests is performed to determine what types of interfaces are the most preferred by end-users. Future works include optimization of the Periscope system architecture (faster response time, better load balance, higher security, etc.) and design of new interfaces, which consists of their graphical design, usability study, and proper inclusion in an exploration path.

REFERENCES

[1] Antarctica homepage: http://www.antarcti.ca/

[2] Benford, B., D. Snowdon, C. Greenhalgh, R. Ingram, I. Knox, C. Brown: VR-VIBE: A Virtual Environment for Co-operative Information Retrieval", Eurographics'95, 30th Au-gust - 1st September, Maastricht, The Netherlands, pp 349-360.

[3] Cat-A-Cone project homepage: http://www.sims.berkeley.edu/~hearst/cac-overview.html

[4] Cugini, J., S. Laskowski, M. Sebrechts: Design of 3D Visualization of Search Results: Evolution and Evaluation, Proceedings of IST/SPIE's 12th Annual Intl. Symposium: Electronic Imaging 2000, San Jose, CA, 23-28 January 2000.

[5] Fekete, J. and Plaisant, C.: Interactive Information Visualization of a Million Items, Proc. of IEEE conference on Information Visualization 2002, Boston, October 2002.

[6] HCIL homepage: http://www.cs.umd.edu/hcil/

[7] Healey, C.G.: Choosing effective colors for data visualization, Proc. of Conference IEEE Visualization '96, pp. 263-270

[8] Hill, B., T. Roger, F.W. Vorhagen: Comparative analysis of the quantization of color spaces on the basis of the CIELAB color-difference formula, ACM Transactions on Graphics, 16(2) 1997, pp. 109-154

[9] InXight homepage: http://www.inxight.com/map/

[10] Mukherjea, S., Y. Hara: Visualizing World-Wide Web Search Engine Results, NEC USA Inc. 1999 International Conference on Information Visualisation July 14 - 16, 1999 London, England

[11] NIRVE project homepage: http://www.itl.nist.gov/iaui/vvrg/cugini/uicd/nirve-home.html

[12] Periscope project homepage: http://periscope.kti.ae.poznan.pl/

[13] Post, D.L., F.A. Greene: Color name boundaries for equally bright stimuli on a CRT: Phase I. Society for Information Display, Digest of Technical Papers 86, pp. 70-73

[14] Rossi, A.M., M. Varga: Visualization of Massive Retrieved Newsfeeds in Interactive 3D, Intl. Conference on Information Visualisation July 14 - 16, 1999 London, England

[15] Sebrechts, M., J. Vasilakis, M. Miller, J. Cugini, S. Laskowski: Visualization of Search Results: A Comparative Evaluation of Text, 2D, and 3D Interfaces, Proceedings of SIGIR'99, pp. 3-10, eds. M. Hearst, F. Gey, and R. Tong, 22nd Intl. Conference on Research and Development in Information Retrieval, Berkeley, California, August 1999

[16] ViOS – how does it works?: http://www.howstuffworks.com/vios.htm

[17] Visualization of a Million Items project homepage: http://www.cs.umd.edu/hcil/millionvis/

[18] VR-VIBE homepage: http://www.emptiness.org/vr/vrvibe.html

[19] Walczak, K., W. Cellary: Building Database Applications of Virtual Reality with X-VRML, Proc. of the 7th International Conference on 3D Web Technology (Web3D-2002), Tempe, Arizona, USA, Feb. 24-28, 2002, pp. 111-120

[20] Ware, C.: Information Visualization, Academic Press 2000

[21] X-VRML homepage: http://xvrml.kti.ae.poznan.pl/