High-Level Visualization of Users' Navigation in Virtual Environments

Lucio Ieronutti, Roberto Ranon, and Luca Chittaro

HCI Lab, Dept. of Math and Computer Science, University of Udine,
Via delle Scienze 206, 33100, Udine, Italy
{ieronutt, ranon, chittaro}@dimi.uniud.it
http://hcilab.uniud.it

Abstract. This paper presents the current status of VU-Flow (Visualization of Users' Flow), a software tool that is able to automatically record usage data in Virtual Environments and provide a set of 2D and 3D visualizations that make it easy for an evaluator to visually detect peculiar users' behaviors and navigability problems. The paper focuses on novel functionalities we recently added to the tool. More specifically, the new version of VU-Flow includes: (i) the possibility of visualizing predominant flow directions for multiple users or multiple visiting sessions, (ii) a visualization aimed at highlighting traffic congestion problems in multi-user VEs, (iii) the possibility of visualizing a replay of users' visits together with audio and video recordings of actual users (e.g. gathered during lab experiments), and (iv) the ability to derive, for each user, a list of quantitative data characterizing her behavior in the VE.

1 Introduction

A commonly used technique to study how people navigate in electronic information spaces is based on recording users' actions and then derive users' navigational patterns. Analyzing users' navigational behavior (e.g., identifying more/less accessed parts of the information space or recurrent navigation patterns) may help one in understanding the effects of design choices. Unfortunately, while several techniques and tools for such activity have been proposed in the contexts of Web sites or, more generally, hypermedia systems (e.g., see [1,2,3]), in the context of Virtual Environments (VEs) there are no established techniques and tools.

In this paper, we present a software tool, called *VU-Flow* (*Visualization of Users' Flow*), which is able to record users' movements in a VE and visualize them using various techniques that help one in deriving information about how users navigate in the VE, highlighting navigability problems, and making hypotheses about interests or preferences of users.

A first prototype of VU-Flow has been presented in [4]. This paper focuses on novel functionalities we recently added to the tool. In particular, the actual version of VU-Flow adds: (i) the possibility of visualizing predominant flow directions for multiple users or multiple visiting sessions, (ii) a novel visualization aimed at highlighting traffic congestion problems in multi-user VEs, (iii) the possibility of visualizing a replay of users' visits together with audio and video recordings of actual

M.F. Costabile and F. Paternò (Eds.): INTERACT 2005, LNCS 3585, pp. 873 – 885, 2005.
© IFIP International Federation for Information Processing 2005

users (typically gathered during lab experiments), and (iv) the ability to derive, for each user, a list of quantitative data (a *feature vector*) characterizing her behavior in the VE, and that can be later studied with other data analysis applications (e.g. spreadsheets or statistical tools) or machine-learning techniques.

The paper is structured as follows. In Section 2, we survey related work; Section 3 explains the main motivations of our research. Section 4 describes the architecture of VU-Flow, while in Section 5 we describe in detail its novel functionalities. Finally, Section 6 concludes the paper by discussing ongoing or future developments of VU-Flow.

2 Related Work

In this Section, we focus on the context of Web/Hypermedia systems, and summarize the main approaches that allow an evaluator to automatically record users' interactions with the system and then analyze them to evaluate usability or characterize users' behavior. In general, usage data can be collected during: (i) a remote evaluation, where users and evaluators are separated in space (and can be also separated in time), or (ii) by carrying out lab experiments, where the evaluator directly observe the user interacting with the system, and may also audio/visual record her.

Two main techniques are available for automatically recording information on users' actions. The straightforward one is based on using Web server logs. This approach, that has the advantage of being inexpensive and applicable to any Web site, has however some disadvantages: only some navigation data can be captured (e.g., in general, if one wants to record the browsing history, it cannot include other Web sites) and the impossibility to capture low-level user interactions with user interface elements (e.g., menus and buttons). A second technique requires to modify Web pages to include code (e.g., javascript and/or Java applets) able to record interface events (e.g., reset of forms, resize and scroll of browser windows); in this case, the acquired data can be sent through the Internet and stored into a remote database.

Starting from recorded usage data, there are mainly three kinds of techniques for deriving usability-related information in the context of Web/Hypermedia applications [5]: *metric-based*, *task-based*, and *inferential analysis*.

Metric-based approaches generate quantitative performance measurements that allow evaluators to identify performance bottlenecks (e.g., slow server response time) that may have a negative impact on the usability of a Web site. However, these approaches focus on server and network performance, and thus provide little insight into the usability of the Web site itself.

Task-based approaches analyze anomalies between the expected users' behavior and what users really do while using the system. For example, the WebRemUSINE tool [1] identifies a usability problem by the lack of correspondence between how users perform a certain task and the model of that task describing how it should be efficiently performed.

Inferential analysis of data includes statistical and visualization techniques to extract usability-related information. Statistical approaches are mainly based on traffic-based analysis (e.g., computing accessed pages per visitor or number of visitors per page) and time-based analysis (e.g., click-streams and page-view durations). By analyzing the number of accesses to Web pages and deriving recurrent patterns of visit, one can derive information on the usability of a Web site. For example, derived patterns may highlight non-optimal (e.g., too long) navigation paths frequently followed by users, and thus suggest design modifications. Some tools (e.g., [2,3,6]) facilitate analysis by also presenting results in graphical formats. For example, the VISVIP tool [2] represents paths followed by users during a site visit by visualizing the Web site as a directed graph (pages are represented with nodes and links with edges), and the paths followed by users through curved lines connecting nodes (different colors are used to represent paths followed by different users). Moreover, VISVIP visualizes the average time spent at each page as the length of a dotted vertical line positioned on the corresponding node. The WebQuilt tool [6] uses a similar graph-based visualization, but employs screenshots of Web pages for the nodes of the graph, and edges are drawn only for traversed links; the thickness of an edge indicates how many times the corresponding link has been followed, while its color indicates the average amount of time spent before clicking the link.

However, graph-based approaches can become visually confusing with large and complex Web sites, or when one wants to follow the usage evolution over time. An interesting visualization technique using 3D graphics to face this problem, called Time Tube [3], uses one or more *Disk Trees* representing the evolution of a Web site structure and usage in time. Time Tube provides the evaluator a sense of direction for users' flow, represents usage evolution in time and identifies significant users' paths.

3 Motivations

The same motivations for studying users' navigational behavior in Web/Hypermedia applications still hold in the context of VEs, since navigation is in both cases one of the fundamental user activities. Additionally, navigation in VEs is generally perceived as a difficult activity by users, and typical navigation problems, such as disorientation and difficulties in wayfinding, are exacerbated by well-known troubles in moving inside a 3D space.

Navigational problems in VE are therefore critical for usability, and typically result in users becoming rapidly frustrated and leaving the VE, missing interesting parts of it, or completing the visit with the feeling of not having adequately explored it.

Although some solutions to improve navigability of VEs (e.g., design guidelines [7] or electronic navigation aids [8,9]) have been proposed, in most cases, finding and correcting navigability problems requires one to observe and analyze how users really interact with the VE. Moreover, unlike in the context of Web/Hypermedia systems (or, more generally, 2D interfaces), there are no extensive and comprehensive sets of verified guidelines for designing navigable VEs.

The techniques and tools for studying the navigational behavior of users in Web/Hypermedia applications cannot be used in the context of VEs, because:

- the structure of the information space is different: while Web sites are mainly a collection of connected information items organized in a graph structure, VE content (e.g., 3D models, images, text and audio) is organized in a 3D space, following a possibly complex spatial arrangement (e.g., the 3D model of a building or an entire city): therefore, the graph-based visualizations mentioned in the previous Section are not effective in the context of VEs.

- while in Web sites users navigate from one item to another by simply selecting the desired one from a set of links, in VEs users typically navigate by continuously controlling the position of their viewpoint through mouse, arrow-keys or 3D pointing devices: therefore, usage data are different and, consequently, the techniques for acquiring them mentioned in the previous Section do not work in the context of VEs.

In the remaining part of the paper, we will propose usage data acquisition and visualization techniques that are effective in the context of VEs, and show how the VU-Flow tool implements them.

4 VU-Flow Architecture

VU-Flow is composed by two modules, called *Data Analysis* and *Visualization* (see Fig. 1). An external module, called *Data Acquisition*, is responsible for collecting navigation data from user's visits to a VE.

More specifically, the Data Acquisition module considers that users move into the VE in a "walking" mode (i.e. the typically adopted navigation modality), and samples a user's position and orientation at brief time intervals. Then, it stores the collected data into a database, together with proper identifiers for the considered VE, user and visiting session (this last data is needed to be able to separately analyze different visits to the same VE).

Since the Data Acquisition module has to capture application-specific events, its actual implementation depends on the technology used to build the VE. The Data Acquisition module we implemented works with VEs built with VRML [10] or X3D [11] technologies, and is also able to send the monitored data through the network, allowing to collect navigation data in the context of remote evaluations.

The Data Analysis module processes the navigation data stored in the database, allowing the evaluator to choose the subset of data to analyze by selecting the VE, set of users and set of visiting sessions of interest. For each VE, the Data Analysis module also needs:

- a *map* of the VE, in the form of a two dimensional matrix where each cell represents a (possibly small) square area of the VE, and where the cell value specifies if the corresponding area can be traveled by users or contains obstacles to navigation (e.g., objects and walls). This map can be automatically constructed from the VE itself, using the method described in [12].
- (optionally) locations in the VE of relevant *Points of Interest* (hereinafter, POIs), e.g. exhibits in a virtual museum or products in a virtual store.

The main task of the Data Analysis module is to compute additional values for the cells of the map that are then taken as input by the Visualization Module, which graphically represents them using two main types of visualizations: (i) by overlaying computed data onto a visual representation of the map (i.e., a 2D visualization), or (ii) by integrating computed data into the VE itself (i.e., a 3D visualization).

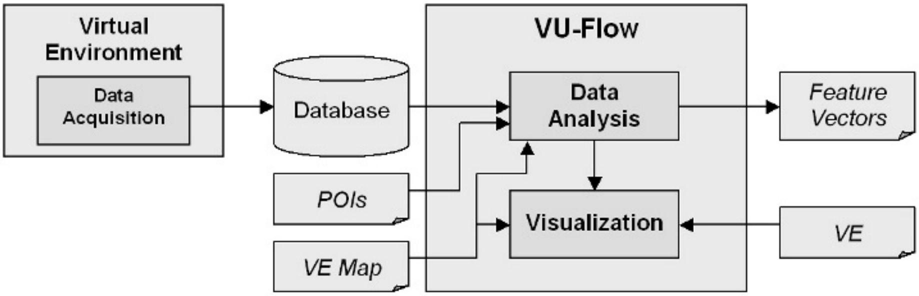


Fig. 1. VU-Flow Architecture

5 Main Functionalities of VU-Flow

VU-Flow provides the evaluator with six main functionalities:

- visualizing (over the map or in the VE), by using different shades of color, more/less traveled areas in the VE or areas where users stayed for more/less time (e.g., see Fig. 2 and 3);
- visualizing over the map the objects in the VE (possibly chosen among the given POIs) that were more/less seen by users during their visit(s) (e.g., see Fig. 4);
- replaying visits of single or groups of users, either employing points and colored lines over the map to draw the user's positions and paths (e.g., see Fig. 5), or visualizing them directly in the VE (e.g., see Fig. 6);
- visualizing over the map, for each part of the VE, the users' predominant flow direction (e.g., see Fig. 7);
- for multi-user VEs, highlighting and visualizing (over the map or in the VE) areas where traffic congestions occurred (see Fig. 8);

The first two functionalities are presented in [4]. The following Sections describe in detail the last three, novel, functionalities.

The new version of VU-Flow includes also the possibility of integration with other analysis and/or classification tools. More specifically, the Data Analysis module is able to derive (and export to a file) a *feature vector* characterizing the behavior of each user by values such as average and standard deviation of the user's speed, average and the standard deviation of the user's angular speed, number of intersections in the navigation path, how much time the user looked toward each POIs.

5.1 Replaying Users' Visits

VU-Flow allows one to replay visits of single or groups of users to a VE and visualize them using either a 2D or a 3D visualization. This functionality can be useful, for example, to review and analyze data collected during an evaluation with users. Since recorded navigational data are just position and orientation samples, VU-Flow employs interpolation techniques to derive navigation trajectories and smoothly replay user's movements.

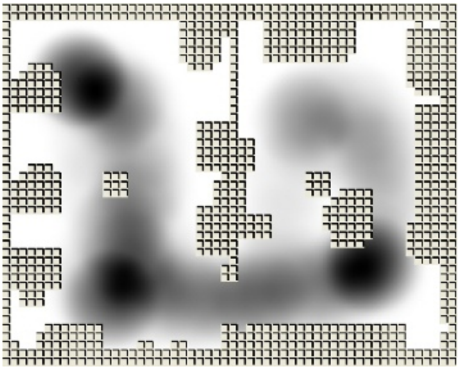


Fig. 2. Visualizing over the 2D map areas where users stayed for more/less time (darker grey highlights areas where users spent more time)

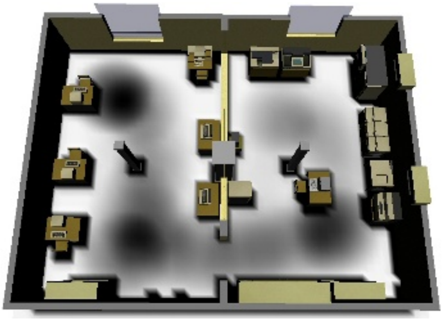


Fig. 3. Visualizing into the VE areas where users stayed for more/less time (darker grey highlights areas where users spent more time)

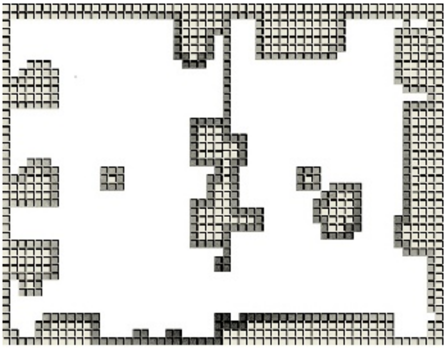


Fig. 4. Visualizing over the map objects in the VE that were more/less seen by users (darker grey highlights more seen objects)

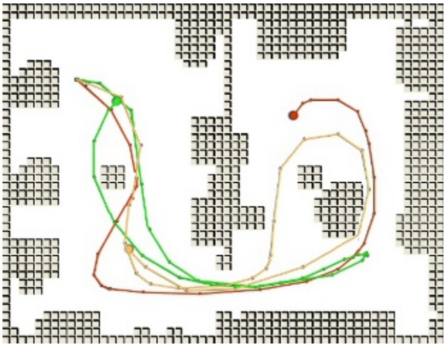


Fig. 5. Drawing on the 2D map paths followed by three users during their visit

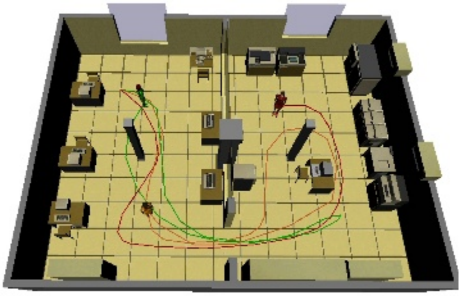


Fig. 6. Drawing into the VE paths followed by three users during their visit

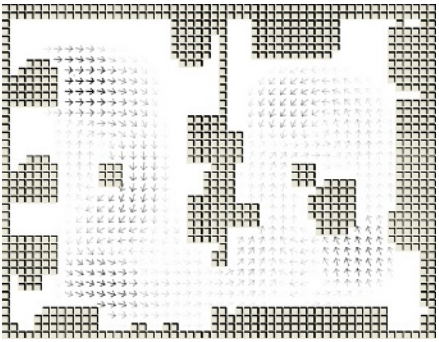


Fig. 7. Visualizing the users' predominant flow directions (arrows indicate flow directions, darker grey arrows indicate more predominant flow directions)

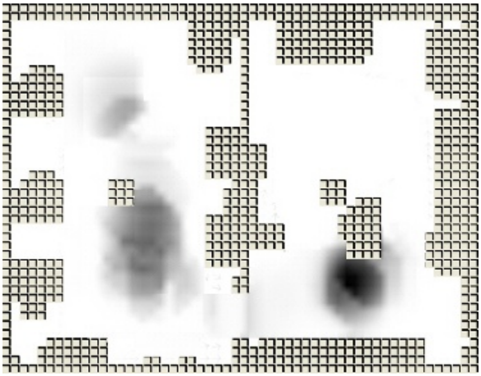


Fig. 8. Visualizing areas where traffic congestion occurred (darker grey indicates more critical traffic congestions)

In the 2D visualization (see Fig. 9a), the current position of the user is drawn on the map as a point, while orientation is represented by two lines that delimit the user's field of view. In this way, the evaluator is both able to observe users' positions in time and estimate where users looked at during the visit. One can choose also to visualize the entire path followed by users up to the current instant (as shown in Fig. 9b).

The 3D visualization replays users' visit into the original VE, by using a different avatar model for each user. The evaluator can observe the replayed visits either through a freely controlled viewpoint (see Fig. 10a), or directly "through the eyes" of one of the users by selecting a specific user's viewpoint (see Fig. 10b). The first solution allows the evaluator to choose the most suitable view of the 3D scene, while the second (that reproduces exactly what a user saw during a visit) may be useful, for example, to precisely identify which objects in the VE have been seen by each user.

To control replay, VU-Flow provides VCR-like controls, i.e. play, fast forward, rewind, pause, and fine tuning of replaying speed (see Fig. 11).

When experiments with users are video-recorded (or only user's voice is recorded, e.g. in a think-aloud experiment), it is possible to replay visits together with video and/or audio recordings. For example, in this way the evaluator can:

- relate users' verbal comments (and/or facial expressions) with interactions in the VE, e.g. to contextualize and disambiguate deictic comments such as "What is this?", "How can I open it?";
- better understand motivations for a particular user's behavior, e.g., when video data are available, in a situation where a user does not interact for many seconds with the application, to point out the real motivation (has the user encountered difficulties in using input devices or was she simply observing something?);
- identify specific areas of the VE where users had navigation difficulties.

Although the Data Acquisition module is not currently able to acquire video and audio data (and thus they have to be separately acquired and then manually synchronized with the navigational data), VU Flow allows the evaluator to display audio and video data during the replay of visits in both the 2D and 3D visualizations (e.g., see Fig. 12). When multiple users' visits are being replayed at the same time, reproducing audio recordings from every user is likely to result in unintelligible sounds. For this reason, VU-Flow allows the evaluator to select which user's audio comments to replay. However, when using the 3D visualization, another interesting option is to use positional audio sources in the VE, where each audio source reproduces comments for a specific user, and is located in the head of the user's avatar. In this case, by moving in the VE, the evaluator may focus on audio comment related with specific users or places, and at the same time perceive comments in the surroundings.

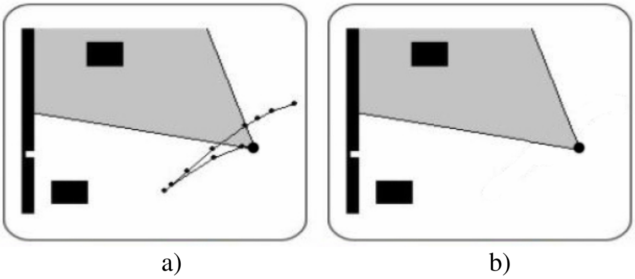


Fig. 9. Replay of user's movements using 2D visualizations

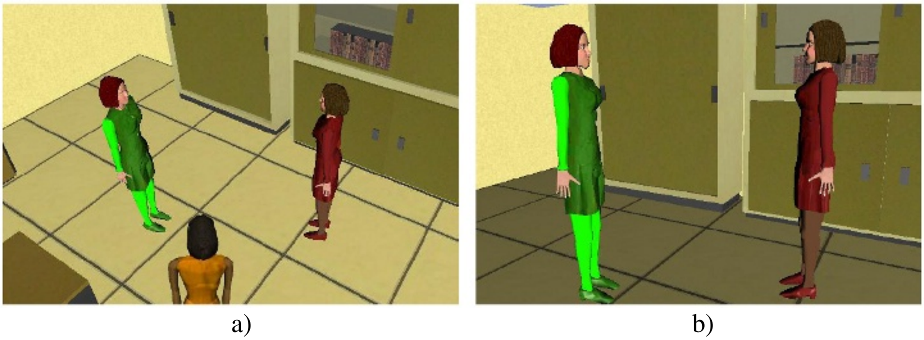
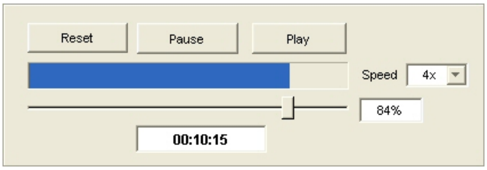


Fig. 10. The same visiting instant seen by employing a freely controlled viewpoint (a) and a user-viewpoint (b)



Speed 4x

84%

00:10:15

Fig. 11. VCR-like controls provided by VU-Flow to control replay

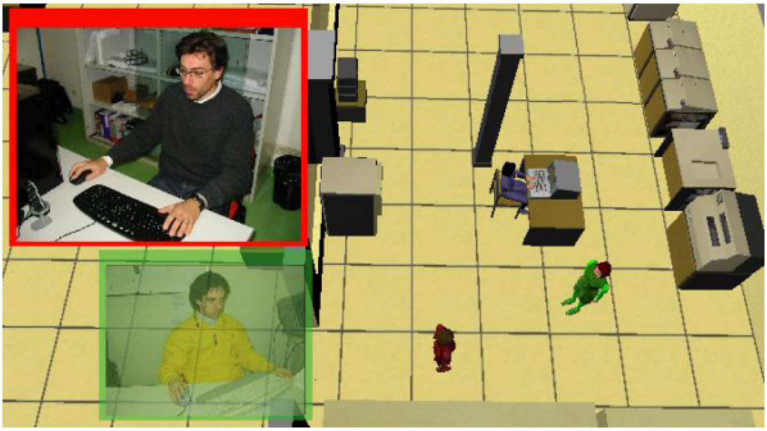


Fig. 12. Replaying visits video recordings of two users

5.2 Deriving and Visualizing Users' Predominant Flows

Visualizing detailed users' paths on the map might help one in identifying user's behavior, but the result is likely to be too visually confusing when one wants to consider a user's very complex and long path, or a population of users (e.g., see Fig. 13) and/or more visiting sessions.

For this reason, in [4], we introduced a set of techniques that, by coloring cells of the 2D map, are able to derive more effective visualizations, e.g. using different colors to visualize areas of the VE where users stayed for more/less time. However, these techniques are not able to visualize users' directions in the visits.

The current version of VU-Flow introduces a new visualization that displays information of predominant flow directions starting from a *vector field* computed by the Data Analysis module. More specifically:

- for each user and visiting session that are under consideration, we first compute a unit vector for each pair of successive navigation samples, where the vector direction is equal to the user movement direction; the computed vector is associated to the map cell where the first of the two navigation samples where taken;
- then, we sum together all vectors associated to the same cell.

As a result, for each cell of the map, the associated vector direction represents the predominant flow direction followed by users in the corresponding part of the VE, and the associated vector length shows the predominance of the computed direction.

VU-Flow visualizes the vector field by employing *hedgehog arrows* [13], a typical technique employed for this purpose. More precisely, an arrow is drawn for each cell of the map, where the arrow color is used to represent the associated vector length (darker means longer), while the arrow orientation corresponds to the associated vector direction. As a result, if an area of a VE was visited by users following a strongly predominant direction, the corresponding arrow will be darker, while the arrow will be lighter if users visited the same area without following a predominant direction.

To better understand the benefits offered by this visualization, in the following we provide a simple example showing parts of different visualizations of the same data recorded on a VE composed by two rooms (denoted by A and B) connected by two passages.

Figure 13 visualizes the recorded data by employing colored lines over the map to show detailed users' paths, Fig. 14 employs different shades of grey to highlight more/less traveled areas, while Fig. 15 visualizes the predominant flow directions. While the first visualization could be used to determine both more traveled areas and movement directions (by observing the recorded data as they are replayed), it is visually quite confusing; the second visualization is more effective in understanding the users' global behavior, since most traveled parts are clearly highlighted, but information on users' movement directions are completely lost. For example, one can determine that the left passage was more traveled than the top one, but cannot determine a predominant flow direction for any of the two passages, nor if the number of times users entered room B is greater than the number of times users exited the same room. By using the novel visualization (Fig. 15), one can easily discover that users predominantly entered room B through the top passage and visited room B following a clockwise direction, while a predominant direction through the left passage cannot be identified. Moreover, the number of times users entered room B is greater than the number of times of users exited the same room.

It is also interesting to note that additional information can be derived by synergically employing both the visualizations in Fig. 14 and 15; in particular, by observing that Fig. 15 does not highlight a predominant flow direction near the left passage, one could hypothesize that only a few users traveled through it; however, by considering also Fig. 14, and noting that the left passage was a highly traveled area, one can rule out the above hypothesis.

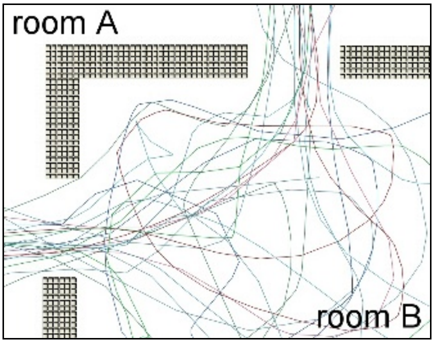


Fig. 13. Detailed users' paths of a population of users

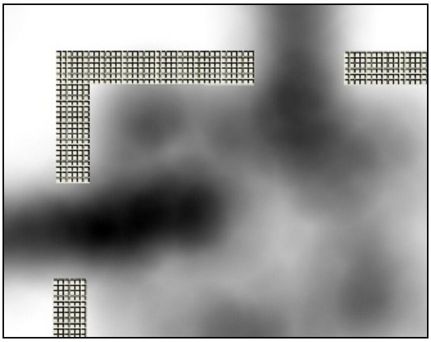


Fig. 14. Areas more/less traveled by users

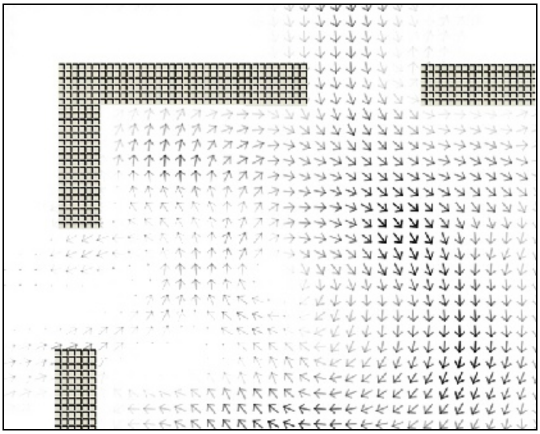


Fig. 15. Users' predominant flow directions (arrows indicate flow directions, darker grey arrows indicate more predominant flow directions)

5.3 Detecting and Visualizing Traffic Congestions

In the context of multi-user VEs, the simultaneous presence of several users navigating can introduce additional navigability issues. For example, navigation difficulties due to traffic congestions may occur when a user's movements are obstructed by the presence and movements of other users. This situation may be problematic when the environment includes narrow passages (e.g., doors or corridors) or when POIs are not effectively spatially distributed in the VE.

The current version of VU-Flow is able to detect and visualize areas where traffic congestions occurred during a multi-user session. Information on traffic congestions is internally represented by associating a real number in [0, 1] to each cell of the map, indicating how much critical a traffic congestion was in the corresponding area of the VE (0 means no congestion).

Intuitively, a traffic congestion occurs whenever an area of the VE is traveled by more than one user in the same time interval (whose length can be changed by the evaluator). Moreover, the criticality of the congestion depends on the number of these users.

More precisely, congestions are identified as follows:

- in a first step, the Data Analysis module groups navigational data such that samples that are mutually sufficiently close in time (i.e., sampling times are within the given temporal interval) and space (i.e., the distance between sampled positions is less than a given value) belong to the same group (note that a sample may belong to different groups). Only groups that contain more than one sample are passed to the second step.
- in a second step, each group is mapped to one or more adjacent map cells and the number of samples in the group is associated to the cell (0 is assigned to cells to which no groups are mapped).

The Visualization module then converts cell values into appropriate shades of a given color. For example, by processing the navigational data used in the previous

Section (see Fig. 13) and considering that they refer to users navigating simultaneously in the VE, the tool derives the visualization depicted in Fig. 16. The visualization uses darker gray to highlight more critical traffic congestion and white to identify no congestion areas. While for the left passage the light-grey areas shows that users traveled the corresponding part of the VE without traffic congestion problems, a serious traffic congestion problem is identified near the top passage. By using also information on more/less traveled areas and predominant flow directions, one can derive that: (i) several users tried to enter room B at the same time through the top passage, and (ii) several users walked through the left passage in opposite directions in different instants (then, without facing traffic congestion problems).

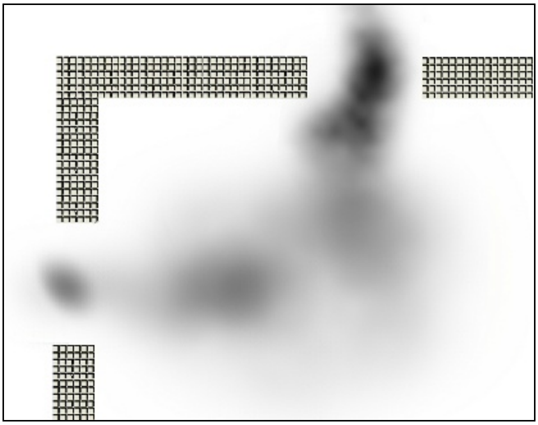


Fig. 16. Visualizing areas of traffic congestion (darker grey indicates more critical traffic congestions)

6 Conclusions

This paper proposed a tool, called VU-Flow, that is able to automatically record usage data in VEs and then visualize them in formats that help an evaluator to understand the effects of VE design on users' navigation.

With respect to future goals of this project, we plan to extend VU-Flow for studying users' behavior in real environments. In this context, the Data Acquisition module should be modified to record information on users' position and orientation through localization technologies, e.g., GPS. It must be noted, however, that while in VEs positions and orientations of users can be precisely determined, in real environments localization technologies usually provide partial and/or inaccurate positioning information.

We are also investigating which kind of feature vectors could be effectively used to identify users' visiting style [14]. To achieve this goal, we are developing a machine-learning system (similar to the one used in [15]) based on the combination of an unsupervised approach (similar vectors are automatically clustered in the same class) and a supervised approach (an expert has to label each vector with a certain class).

Acknowledgments

Our research has been partially supported by the Italian Ministry of Instruction, University and Research (MIUR) under the project "Web-based management and representation of spatial and geographical data", subproject "User Interfaces for the Visualization of Geographical Data on Mobile Devices".

References

1. Paganelli, L., Paternò, F.: Intelligent Analysis of User Interactions with Web Applications. In: Proc. Intelligent User Interfaces (IUI 02), ACM Press, New York (2002) 111-118
2. Cugini, J., Scholtz, J.: VISVIP: 3D visualization of paths through web sites. In: Proc. Database and Expert Systems Applications (DEXA 02), IEEE Computer Society Press (2002) 259-263
3. Chi, E., Pitkow, J., Mackinlay, J., Pirolli, P., Gossweiler, R., Card, S.: Visualizing the Evolution of Web Ecologies. In: Proc. Conference on Human Factors in Computing Systems (CHI 98), ACM Press, New York (1998) 400-407
4. Chittaro, L., Ieronutti, L.: A Visual Tool for Tracing Behaviors of Users in Virtual Environments. In Proc. Advanced Visual Interfaces (AVI 04), ACM Press, New York (2004) 40-47
5. Ivory, M., Hearst, M.: The state of the art in automating usability evaluation of user interfaces. ACM Computing Surveys (CSUR), ACM Press, New York (2001) 470-516
6. Hong, J., Landay, J.: Webquilt: A Framework for Capturing and Visualizing the Web Experience. In: Proc. World Wide Web. ACM Press, New York (2001) 717-724
7. Vinson, N. G.: Design Guidelines for Landmarks to Support Navigation in Virtual Environments. In: Proc. Conference on Human Factors in Computing Systems (CHI 99), ACM Press, New York (1999) 278-284
8. Chittaro, L., Burigat, S.: 3D Location-pointing as a Navigation Aid for Virtual Environments. In: Proc. Advanced Visual Interfaces (AVI 04), ACM Press, New York (2004) 267-274
9. Darken, R. P., Cevik, H.: Map Usage in Virtual Environments: Orientation Issues. In: Proc. IEEE Virtual Reality 99, IEEE Computer Society Press (1999) 133-140.
10. VRML. ISO/IEC 14772-2:2004, http://www.web3d.org/x3d/specifications/vrml/ISO-IEC-14772-IS-VRML97WithAmendment1/#ISO_IEC_14772Part2 (last access on April 2005).
11. X3D. ISO/IEC FDIS 19775, http://www.web3d.org/x3d/specifications/index.html (last access on April 2005).
12. Ieronutti, L., Ranon, R., Chittaro, L.: Automatic Derivation of Electronic Maps from X3D/VRML Worlds. In: Proc. International Conference on 3D Web Technology, ACM Press, New York (2004) 61-70
13. Post, F., van Wijk, J.: Visual Representation of Vector Fields: Recent Developments and Research Directions. In: Scientific Visualization: Advances and Challenges, Academic Press, London (1994) 367-390
14. Veron, E., Levasseur, M.: Ethnographie de l'Exposition. Bibliothque publique d'Information, Centre Georges Pompidou, Paris (1983)
15. Sas, C., O'Hare, G. M. P, Reilly, R.: Virtual Environment Trajectory Analysis: A Basis for Navigational Assistance and Scene Adaptivity. Future Generation Computer Systems, Special Issue on Interaction and Visualisation Techniques for Problem Solving Environments, Elsevier (2004)