Virtual Prints: An Empowering Tool for Virtual Environments

A. Mourouzis[1], D. Grammenos[1], M. Filou[1], P. Papadakos[1], C. Stephanidis[1,2]

[1]Foundation for Research and Technology Hellas
Institute of Computer Science
Science and Technology Park of Crete
Heraklion, Crete, GR-71110 Greece
{mourouzi, gramenos, filou, papadako, cs}@ics.forth.gr

[2]University of Crete, Department of Computer Science

Abstract

The concept of Virtual Prints (ViPs), as digital counterparts of real-life tracks that people leave behind, has been introduced for supporting navigation, orientation and wayfinding in Virtual Environments (VEs) and has been explored using a prototype VE equipped with a simplified ViPs mechanism. This paper describes an elaboration of the ViPs mechanism with the aim to support a number of functions, popular, if not standard, in conventional applications, but also useful and required in VEs.

1 Introduction

In (Grammenos, Filou, Papadakos & Stephanidis, 2002), the concept of Virtual Prints (ViPs), as the digital counterparts of real-life tracks that people leave behind, was introduced for supporting navigation, orientation and wayfinding in Virtual Environments (VEs). ViPs can be manifested in three different types (Grammenos et al., 2002): (a) while 'inhabitants' of a virtual world are moving, they are leaving behind their Virtual Footprints (ViFoPs); (b) every time they are interacting ('contacting') with a VE their Virtual Fingerprints (ViFiPs) are 'imprinted' on it; and (c) Virtual Fossils (ViFossils) are special marks that can be permanently left within the virtual space, or on any object upon user request, and can be considered as a kind of personal landmark. Following the preliminary investigation that had been carried out using a prototype VE equipped with a simplified ViPs mechanism, this paper proposes a further elaborated ViPs concept and mechanism that can be employed to support not only navigation, orientation and wayfinding in VEs, but also an additional number of functions, popular, if not standard, in conventional applications, but potentially useful also in VEs, such as interaction shortcuts, bookmarks, help support, interaction history facility, back / forward facility, undo / redo and repeat facility, annotation facility, and facility for highlighting content or marking / identifying (non) visited areas.

2 Elaborated ViPs Properties & Characteristics

Just like footprints in the real world, ViFoPs can be visualised in VEs and thus provide a continuous (i.e., snail tracks like) or discontinuous (i.e., dashed like) three dimensional (3D)

representation of the path followed by any user moving in the virtual space. ViFoPs can be depicted in various ways, depending mainly on the characteristics of the application and on the user's requirements and preferences. For a discontinuous representation of the user's path, ViPs can be depicted as simple 3D objects (e.g., cones which may also provide information regarding the user's orientation), or more realistic 3D objects (e.g., a 3D model of a shoe's sole). To make the path more distinguishable, *Connecting Lines* (e.g., thin 3D cylinders) between subsequent ViFoPs can be displayed upon user request (see figure 1), which is a similar technique to the one used when a continuous representation of the user path is desired. A continuous path can be depicted as a beeline using ViFoPs as edges and *Connecting Lines* as connectors. To increase the curvature of a continuous representation, and thus achieve a better graphical representation of 'continuity', the number of edges (i.e., the number of ViFoPs released per unit of time or space) needs to be increased. Alternatively, a number of intermediate *Nodes,* a simplified version of ViFoPs that can only store position co-ordinates x, y and z, can be used.

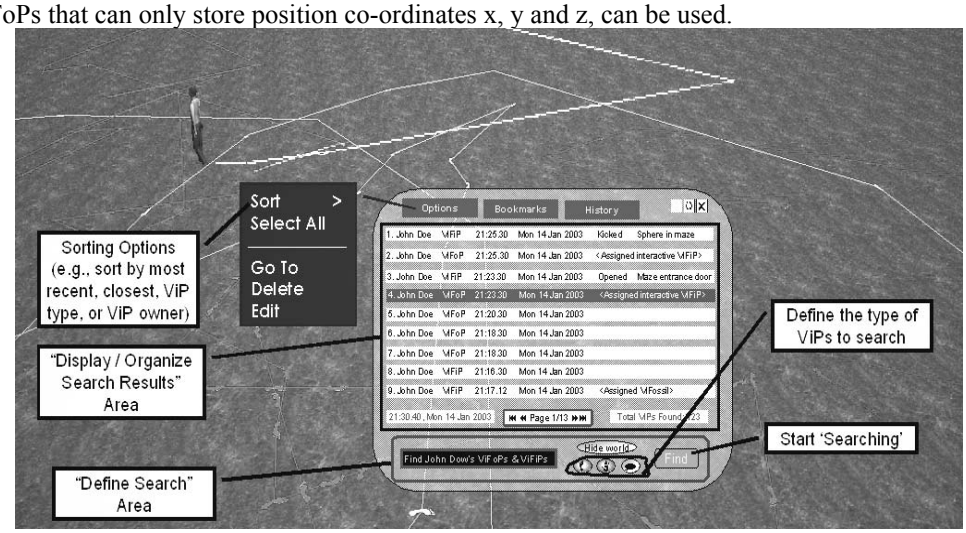

Figure 1: ***Search Console*** **(the Connection Lines between ViFoPs are also visible)**

Each ViFoP can store and therefore provide: (i) spatial information, i.e., position and orientation of the user in the virtual world; (ii) chronological information, e.g., creation, last accessed, or last modified time and date; (iii) personal data information, e.g., owner name or depiction of his/her ViFoPs; and (iv) information about assigned ViFiPs or ViFossils. Each ViFiP is always assigned to one and only ViFoP that is automatically released to record and represent the position and orientation of the user at the moment the interaction took place, just like each ViFossil is assigned to one and only ViFoP to record the position and orientation of the user at the moment the ViFossil was released. This way, a VE user may retrieve information regarding non-visited sites, including undiscovered options, through the ViFoPs of other user(s). Furthermore, a *Connecting Line* between the ViFoP and the assigned ViFiP / ViFossil can be drawn to make the relation visible. ViFoPs can be released anytime, upon user demand or automatically: (a) at specific time or / and space intervals; (b) each time a ViFiP is released; and (c) each time the user releases a ViFossil. Finally, the concept of Virtual Path (ViPath) can be introduced to group ViPs released by a single user that share common spatial (e.g., ViFoPs left in a specific virtual area), chronological (e.g., ViFoPs left during a specific day), or other (such as semantic, e.g., ViFoPs left while performing a specific task) characteristics. ViPs that belong to the same ViPath may 'inherit' part of the information stored in the 'parent' ViPath rather than store it themselves.

As mentioned above, the idea behind ViFiPs is that every time VE 'inhabitants' interact with a VE, their ViFiPs are 'imprinted' on it. Thus, ViFiPs are released automatically: (a) each time an interaction between the user and the VE is detected; and (b) each time the user collides with a component of the virtual world. Whenever possible, the ViFiP is 'imprinted' on the corresponding part of the virtual world. For example, in case of interaction using a pointing device, the user's ViFiPs can be 'imprinted' and thus visualised at the pointing area, e.g., as a cube or, more realistically, as a 3D model of a fingerprint. Non-visual ViFiPs (e.g., ViFiPs generated from speech-based interaction) can also be released, and it is the responsibility of the assigned ViFoP to make apparent the existence of the ViFiP in question. Thus, ViFiPs can be visual or non-visual and store: (i) spatial information when applicable, i.e., co-ordinate of the position and orientation of the interaction; (ii) personal data information, e.g., depiction of this ViFiP; (iii) elucidatory information about the interactive component, e.g., name of type of the object such as "Media Laboratory door"; (iv) descriptive information about the performed user action, e.g., "Left-click on the interactive device"; (v) other information about the VE reaction, such as semantic information e.g., "The Media Laboratory door opened".

ViFossils are special permanent marks that can be: (a) left anywhere within the virtual world; (b) 'pinned' (i.e., attached) to a specific virtual object; or (c) applied to a virtual object, e.g., by applying a specific texture on it. ViFossils can store: (i) spatial information, i.e., the coordinates of the ViFossil's position and orientation in the virtual world; (ii) personal data information, e.g., style and appearance of the ViFossil; (iii) a message in any digital form such as text, audio, or multimedia, e.g., voice-delivered instructions of use of an interacted component; and / or (iv) elucidatory information about the surrounding context of the ViFossil, e.g., "Left by the Media Laboratory door". As a result, ViFossils depending on the message may also be employed to provide or retrieve help support. For example, a number of ViFossils may be attached to specific components of a VE, providing related descriptions and / or guidance for inexperienced users. Just like ViFiPs, ViFossils can be visual or non-visual, and visual ViFossils can be depicted in various ways depending mainly on the characteristics of the application and on the user's requirements and preferences. Indicatively, ViFossils can be depicted as pins, yellow stickers, road signs, wall signs / posters, or pets. Furthermore, ViFossils can also be stored and retrieved through classic 'Favorites' like mechanism. Upon user's request, ViFossils can be grouped in folders. Each folder can be named with a significant title and hold a number of ViFossils and / or other sub-folders.

3 Interacting with ViPs

The ViPs mechanism (automatically) records a sequence of user actions by generating ViPs and thereby storing interaction history related information. A user interacting with a VE needs to be able, at any point in time, to activate or deactivate the ViPs mechanism aiming, for example, at memory space saving. Moreover, once the ViP mechanism is active, there are a number of options related to ViPs that VE users need to have access at any point in time, such as:

- Release a new ViP (e.g., a ViFossil).
- Start or delete a ViPath.
- Perform *ViPs-based Navigation & Interaction*. ViPs can support, among other things, synchronous and asynchronous *social navigation* in collaborative VEs (Grammenos et al, 2002). For example, it is possible to automatically follow another leading VE user / companion, which allows the user to focus on other tasks rather than on navigation. In addition, a VE user may 'take a shortcut' (i.e., be 'teletransported') to another of his / her ViP, such as to previous or next ViP (i.e., a 'Back & Forward' effect) or straight to

his first ViP (i.e., a 'Go to Home' effect). In a similar way, since ViFiPs hold information regarding the user interactions with components of VEs, ViPs may also support 'Undo', 'Redo', and 'Repeat' as well as *object-focused interaction* (Hindmarsh et al, 2000). It is also feasible to integrate such ViPs-based navigation and interaction functionality into a single component such as a *Navigation & Interaction Console* (e.g., see figure 2).

- Conduct *ViPs-based Search*. ViPs can provide, among other things, useful feedback regarding the position, time and nature of previous (inter-) actions of VE users. Such information can be visualised in a VE by the means of ViPs, but also accessed by the means of (2D) lists and catalogues. For instance, VE users can browse or search (for instance through keywords search or mixed search combining specific time or ViP type constrains) available lists of ViPs for specific information. This way, the ViPs mechanism may also support *navigation by query* (van Ballegooij & Eliëns, 2002) in VEs. This type of ViPs - related functionality can be integrated into a single component, such as a *Search Console* (e.g., see figure 1).
- Perform *ViPs-based Measurements*. ViPs hold information about their actual position in the virtual world, in terms of their co-ordinates in the three dimensions (x, y, z), and thus allow the automatic calculation of distances among them. In a similar way, since ViPs also store chronological information, such as creation or last accessed date and time, they allow, among others, the automatic calculation of the time required to navigate from one place to another or to perform a specific sequence of actions.

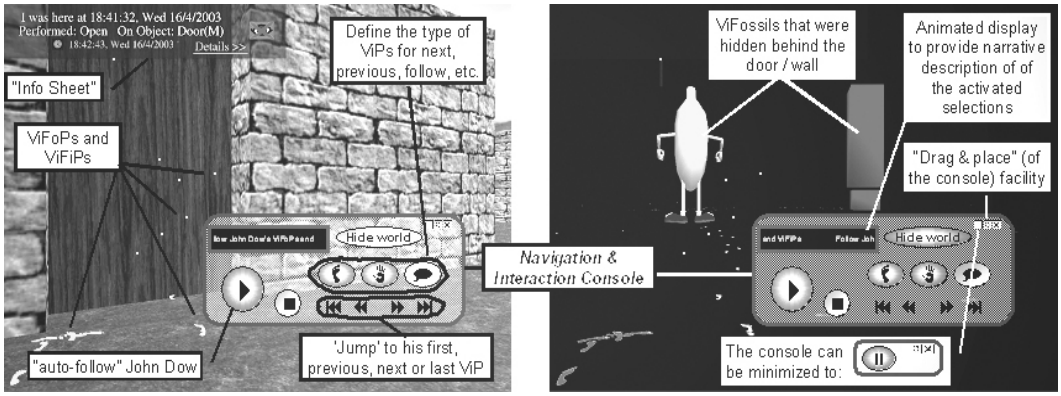

Figure 2: ***Navigation & Interaction Console*** **(right in 'Only ViPs View" mode)**

- Access *ViPs-based Interaction History*. Since the ViPs mechanism records information regarding the user's past navigation and interaction steps, it can also be utilized as an interaction history mechanism. For instance, a VE user may access a list of his / her previous actions performed during a specific day by revisiting the corresponding ViPath or by browsing / searching the corresponding lists of ViPs. It is feasible to incorporate such history-related functionality into the *Search Console* mentioned above.
- Access *ViPs-based Bookmarks*. Once again it is possible to incorporate any bookmarks- related functionality into the *Search Console* mentioned above.
- Access the *Configurations of ViPs*, such as: (i) modify the way ViPs are depicted; (ii) hide / display ViPs, while the ViPs mechanism is active; (iii) reduce or increase the number of displayed ViPs on a percentage basis, e.g., the user may choose to limit the

volume of ViPs by displaying only 60% of the recorded ViPs; (iv) hide or display the Connecting Lines; (v) interchange between 'Only ViPs' (i.e., ViPs 3D map, see figure 2, right part) and normal display (see figure 2, left part); (vi) scale ViPs up or down, i.e., an 'inflate' / 'deflate' effect; (vii) personalise or share own ViPs by specifying whether these are personal, 'read only', or 'read & write'; (viii) modify the way ViPs are depicted; or (ix) edit time – related configurations of ViPs, e.g., modify their time-sensitivity by increasing their lifespan.

According to (Grammenos et al, 2002) each ViP stores directly or indirectly significant user-related information (e.g., spatial, chronological, personal data) that can be displayed upon user request (e.g., when the user 'points' a specific ViP) through an *information sheet*. Furthermore, each ViP can be selected offering a number of alternative options, such as: (a) move, (re-) orient, bookmark, or delete the ViP; (b) 'select' the parent ViPath (i.e., the ViPath to which the selected ViP belongs) and thereafter copy & paste or bookmark its ViPs; (c) 'select' the owner, and thereafter access his / her *ViPs-based Interaction History* or perform a number of application dependent (i.e., ViPs independent) tasks such as block, talk to, or email, the owner, etc.

4 Conclusion & Future Work

ViPs, are 'inspired' from the real world, are easy to understand, and as digital entities, can be subject to processing, and can acquire a number of attributes, e.g., personal or shared, visible or hidden, interactive, dynamic, accessible, etc. A number of user tests are planed to formally assess the usability of the evolved ViPs mechanism (effectiveness, efficiency and user satisfaction) in supporting, not only navigation, orientation and wayfinding in VEs, but also a number of functions, commonly found in conventional applications, but also useful and required in VEs. Further work on ViPs currently under way concerns: (a) the identification of potential, e.g., in evaluation as a review tool; (b) the support of VE designers and developers by delivering ViPs-related guidelines for developing the ViPs mechanism; and (c) creating a re-usable ViPs mechanism to be used as a plug-in to VEs, so that ViPs can be easily adopted by the broader VE users and / or VE developers community.

Acknowledgments

Part of the work presented in this paper has been carried out in the context of "VIEW of the Future" (IST-2000-26089) project, and funded by the European Commission in the framework of the Information Society Technologies (IST) Programme.

References

Hindmarsh, J., Fraser, M., Heath, C., Benford, S., & Greenhalgh, C., (2000). Object-focused interaction in collaborative virtual environments. In *ACM Transactions on Computer-Human Interaction (TOCHI)*, v.7 n.4, p.477-509, Dec. 2000.

Grammenos, G., Filou, M., Papadakos, P., & Stephanidis, C., (2002). Virtual Prints: Leaving trails in Virtual Environments. In Proc. of the Eighth Eurographics Workshop on Virtual Environments, Barcelona, Spain, 30-31 May.

van Ballegooij, A.,& Eliëns, A., (2001). Navigation by Query in Virtual Worlds. In Proc. of Web3D 2001 Conference, Paderborn, Germany, 19-22 Feb 2001.