VR4U2C:
A Multiuser Multiperspective Panoramic Browser
Using QuickTime VR and Java
Featuring Multimonitor and Stereographic Display

Noor Alamshah Bolhassan, William L. Martens, and Michael Cohen
Spatial Media Group, University of Aizu
Aizu-Wakamatsu 965-8580
Japan
e-mail: m5041124, wlm, mcohen@u-aizu.ac.jp

Abstract

One of the first principles of immersive visual display is to fill the user’s field of view with imagery, often of a single environment, sometimes an actual place remotely viewed in space and/or time. The means to project such imagery depends upon both the display devices and the software environment supporting image delivery to the display devices. We have developed a multiuser multiperspective panoramic browser using Apple’s QuickTime VR technology and the Java programming language, together with the support of the QuickTime for Java application programming interfaces (API). This unique QTVR browser allows coordinated display of multiple views of a virtual environment, limited primarily by the size and number of monitors or projectors assembled around individual or groups of users in various viewing locations. Named “VR4U2C” (“virtual reality for you to see”), the browser, developed by the first author for both Macintosh and Windows operating systems, is one of many integrated clients in the University of Aizu Spatial Media Group’s Multimodal Groupware suite— which includes the Internet Chair, 2.5D Dynamic Map, PSFC Proxy, Hero Telerobot, Helical Keyboard, Soundscape-Stabilized (Swivel-Seat) Spiral-Spring, and iAppli “icon” graphic interface for mobile telephone— and interoperates seamlessly with them. VR4U2C can be used interactively to explore and examine detailed multi-dimensional, virtual environments (photorealistic or otherwise) using a computer and conventional input devices—including mouse, trackball, track pad, and keyboard. In addition, it provides a unique solution to the problem of interactive stereoscopic display of QTVR imagery.

VR4U2C has several novel features, including stand-alone and client/server modes, and multiple synchronized windows and monitors with the same or different viewing angles. Synchronization via a Java-implemented client/server (c/s) framework enables multi-window and -monitor configurations of QTVR browsers, even when the windows and monitors are discontinuous, suggesting the idea of “panoramic panoramas,” panoramic viewers that piecewise partially or totally surround a user. The further ability to intelligently and seamlessly switch panoramic scenes enables several novel QTVR features, including stereographic viewing. Users can utilize VR4U2C for viewing a stereo panoramic scene using special eyewear and several well-known viewing techniques such as parallel, cross-eyed and over/under viewing techniques. In addition, a stereo panoramic scene can be projected through polarized filters using two projectors onto a silver screen. Then users can view this stereo effect through passive polarized eyewear.

Keywords: multiuser, multiperspective, multi-display, panoramic scene, stereographic QTVR, image-based rendering (IBR).

1. Introduction

This paper describes a novel Java-based browser that enables multiple users to enjoy panoramic views of one or more environments via Apple’s QuickTime VR technology. Whether the visual environment to be displayed is computer generated or photographically captured, the browser, named “VR4U2C” (“virtual reality for you to see”), can immerse a viewer or viewers in that environment using a wide variety of spatial configurations of visual display devices. Fig. 1 depicts a deployment that might be called a “video-wall” configuration. The viewer in this figure (rendered in wire-frame) is surrounded by a natural scene (perhaps captured using a camera fitted with a fish-eye lens), and that video wall is recognized only by the monitor frames. What is unique about the VR4U2C browser is that it provides several features beyond those provided by ordinary video-wall displays. First, one viewer’s interactive control over viewer perspective can be synchronized across multiple sites for other
viewers. Each viewing site can have its own unique spatial configuration of display devices, some of which providing immersive stereographic imagery. So, in contrast to the more conventional idea for a video wall depicted in Fig. 1, the flexibility of VRβUβC suggests the idea of "panoramic panoramas" that piecewise partially or totally surround a user. Multiple synchronization of windows and monitors with the same or different viewing angles is accomplished seamlessly for each user via a Java-implemented client/server (c/s) framework, even when the windows and monitors are discontinuous. This paper focuses upon only a few potential display-device configurations, in order to explain implementation details and functions of the browser, beginning with an introduction to basic concepts in spatialized hypermedia, including panoramic and stereographic display.

![Image](image_url)

Fig. 1: Graphic depiction of a curved "video-wall" configuration of visual display devices showing immersive QTVR imagery to a viewer. (Graphic produced by "Eyes, Japan.")

1.1 Degrees of Freedom and Hypermedia

There are several dimensions of spatial motion that can be experienced in the "real world," including translational degrees-of-freedom (sway, heave, and surge), and rotational degrees-of-freedom (roll, pitch, and yaw). The dimensions can also describe sound source localization (azimuth, elevation, and range/distance).

1.2 Hypermedia and Stereo Panoramas

Hypermedia combines a hypertext system and media elements (or multimedia, including sound, graphics and images, audio, video, animation), and makes use of a variety of interactive tools to tie them together in a meaningful way. Some people also refer this term to virtual reality, artificial reality, cyberspace, and so on.

The word "panorama" was introduced in 1792 in London to describe a large cylindrical room entirely covered by a 360° painting [Na97] [Na91] [BK01]. However nowadays, in addition to static panoramas, we can also create and display complete 360° scenes over time as "moving panoramas." Computer-based moving panoramas—such as Apple’s QuickTime VR1 [Kit98], Helmut Dersch’s PTViewer², and iPIX Viewer³—can be produced by tiling multiple rectangular images or by combining images captured by a fisheye lens. Such systems allow a viewer to look around (angular movement—i.e., panning and tilting), but not to move around (translational movement—i.e., dollying and tracking). The zoom angle can determine how much of the source image is in the playback window. Zooming changes the field of view (FoV), the visual angle subtended in the display.

The reason that people have stereoscopic vision is because we have two eyes. The lenses of these two eyes project two slightly different images onto the retinas, which are then transformed, by the brain, into a spatial representation [Na97]. An ordinary panorama represents only a single point of view, but a stereo panorama requires more than two points of view. In this case, we use multiple panorama movies of the same scene taken from slightly different point of view (separated by an interocular distance). Then using one of many viewing technologies, the stereoscopic movies are combined in our mind producing a third dimension of depth, as illustrated by Fig. 2.

![Image](image_url)

Fig 2: Depth: 3rd Dimension of Stereo Images

---

1 www.apple.com/quicktime/qtvr
2 www.fh-furtwangen.de/~dersch/PTVI/doc.html
3 www.ipix.com
2. Aim and Objectives
This research involved development of a multiuser multiperspective panoramic VR (QTVR) technology and the Java programming language [BC01].

The main objectives of the research are:
- To clearly identify and understand methods for capturing panoramic scenes, producing QTVR movies.
- To develop a Java program that can be used to interactively explore QTVR movies with other users in real-time through network.
- To integrate QTVR movies seamlessly with other clients in the Multimodal Groupware suite [KCNH01], including the Internet Chair [Coelho00] [KCA00], 2.5D Dynamic Map [MSC01], PSFC Proxy [AMY+98], Hero Telerobot [HOS97] [HST99] [YCHY01], Helical Keyboard [KCO1] [NC01b], Soundscape-Stabilized (Swivel-Seat) Spiral-Spring [CS00], and iAppli “iCan” application for mobile phones [NC01a].
- To elaborate more interesting features for viewing QTVR movies on computer and other equipment such as TV screens, head-mounted displays (HMDs), etc.

3. Related Systems
3.1 Panoramas
Many advanced panoramic viewers have been developed recently, including the well-known Elumens VisionDome,4 which can deliver a full-color, high-resolution, raster-based, interactive, 3D display, with a 360° by 180° panorama projection on the interior of a hemispherical dome. A dozen or more people can simultaneously participate and collaborate without having to use restrictive head-mounted displays or goggles. The tilted hemispherical screen is positioned so as to fill the field-of-view of the participants, creating a sense of immersion in the same way that a large-screen cinema draws its audience into a scene. The observer loses most of the cues regarding display surface position, such as screen edges, and perceives 3D objects beyond the surface of the screen. The dome itself allows freedom of head motion, so that the observer can change direction of view, and yet still have vision fully encompassed by the image.

3.2 Stereograms
The Simalabim Cyberscope is an optical hood that can be mounted on monitors of various size. Stereo pictures are drawn sideways. The left-eye image is drawn on the left half of the screen up to the left. The right-eye image is drawn on the right half of the screen with up to the right.

4. Implementation Details
4.1 Panoramic Capture
We photographically capture panoramic scenes using the Nikon5 CoolPix990 digital camera, its 8 mm Fisheye Converter FC-E8, Kaidan6 KiWi:90 rotating (180° detented) tripod head, and any standard tripod, and then combine multiple fisheye images into integral panoramic scenes in post-production using Adobe Photoshop, QuickTime VR Authoring Studio,7 DeFish,8 Panorama Tools,9 Ptegui,10 SoundSaVR,11 QTVR MakeCubicPPC,12 and VR Worx13 stitching and authoring software.

4.2 Stereo Panoramic Capture
The equipment used for capturing and preparing stereo panoramic scenes includes all that for ordinary panoramic plus a Jasper1+ 8° stereo slide bar. Refering to Fig. 3, firstly, place a camera at the center of slide bar and take a picture, capturing a hemisphere. Then without moving the slide bar, rotate the camera 180° and take another picture. These two images are stitched together using some combination of authoring and stitching software into a panoramic scene, which can later be designated as the left-eyed movie.

To make the right-eyed side of a panoramic scene, take another picture after moving the camera an interocular distance (nominally 65 mm, or about 7 cm) towards the right of slide bar. After that, rotate the slide bar 45° and take another picture. Rotate it again and take the third picture. Repeat this until all necessary pictures (eight altogether) have been captured. Afterwards, convert eight fisheye images to rectangular images, and then stitch or combine these rectangular images into a panoramic image, which can be designated as the right-eyed movie. Finally, this pair of panoramic scenes (JPEG format) can be converted to QTVR movies, using QTVR MakeCubicPPC or VR Worx.

4.3 QTVR Movie Callbacks
Callbacks are used to allow QuickTime-triggered invocation of developer-supplied subroutines, QuickTime calling back into Java through the movie controller, movie, and QuickTime VR APIs [MS99]. These callbacks are used in VRU:U:C to perform tasks when certain conditions arise within QuickTime itself. MovieDrawingComplete callback is used to notify a supervising program whenever QuickTime has drawn to the screen. Once a movie draws on

---

4 www.elumens.com

---

5 www.nikon.com
6 www.kaidan.com
7 www.apple.com/quicktime/qtvr/authoringstudio
8 www.iqtvra.org/DeFish
9 www.sf-furtwangen.de/"derech
10 www.ptegui.com
11 www.smgvrx.com
12 developer.apple.com/quicktime/quicktimeintro/tools
13 www.vrtoolbox.com
14 www.stereoscopy.com/jasper
the screen, an instance of the subclass (MovieDrawing) implementing this callback will be called automatically, and will compute wide view (W) and full pan view (F) for calculating pan angle (PA) of other windows, as shown in Fig. 4.

Moreover, there are two more QTVR callbacks provided in QuickTime VR: QTVRInterceptor for panning, tilting and zooming processes, and TVREnteringNode for entering node processes. The subclass implementing the QTVRInterceptor callback contains a number of useful methods, as outlined in Table 1.

<table>
<thead>
<tr>
<th>Method</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>getPanAngle()</td>
<td>To get/set current pan angle</td>
</tr>
<tr>
<td>setPanAngle()</td>
<td></td>
</tr>
<tr>
<td>getTiltAngle()</td>
<td>To get/set current tilt angle</td>
</tr>
<tr>
<td>setTiltAngle()</td>
<td></td>
</tr>
<tr>
<td>getFieldOfView()</td>
<td>To get/set current field of view</td>
</tr>
<tr>
<td>setFieldOfView()</td>
<td></td>
</tr>
<tr>
<td>ptToPanAngle()</td>
<td>To get pan angle at certain point (in pixels) on movie window</td>
</tr>
<tr>
<td>gotoNodeID()</td>
<td>To move to another node in multinode movie</td>
</tr>
</tbody>
</table>

Table 1: Relevant Callback Methods

4.4 Multithreaded Implementation

The benefits of multithreading are better interactive responsiveness and real-time behavior. [HC01, p.10] Upon opening a movie, VR4U2C will create a thread for its own operation, and other threads for displaying each window/frame, as shown in Fig. 5. Every process in the opened movies can be run in parallel on the same machine.

4.5 Groupware Architecture

Fig. 6 illustrates the relationship between VR4U2C and other clients in our Spatial Media Group’s Multimodal Groupware suite.

As one of many integrated clients, VR4U2C connects to a common server to exchange parameters with other clients synchronously. Conforming to our groupware protocol by implementing the CVEClientIF interface (abstract superclass) provides get methods. Set methods are inCVEClient class, an instance of which is linked to our application. Upon receiving values using getOrientation() method, VR4U2C will assign pitch to tilt angle value, and yaw to pan angle value. However, the program will only save the roll value. Afterward, if the user changes tilt angle or pan angle, these three values will be multicast through the server using setOrientation() method. Furthermore, VR4U2C uses the getExtraParam() method to get “zoom” value from the server, assigning it to the field-of-view data-field. Symetrically, if this variable is updated, it will be sent to the server using setExtraParam() method.
Fig. 4: Calculation for Wide View, Mullion Wide View, and Pan Angle

\[ W = 2 \times [ \text{PA at (0, 0)} - \text{PA (A)} ] \]

\[ M = (\text{mullion / width}) \times W \]

\[ \text{PA (B)} = \text{PA (A)} - (W + M) \]

\[ \text{PA (C)} = \text{PA (A)} - 2 \times (W + M) \]

Fig. 5: Example of Multithreaded Execution

\[ \text{VR}_4 \text{U}_2 \text{C} \]

Pan angle, tilt angle, and field of view

Movie 1

Movie 2

Movie 3

\[ \mathbf{W}_{11} \quad \mathbf{W}_{12} \quad \mathbf{W}_{21} \quad \mathbf{W}_{31} \quad \mathbf{W}_{32} \quad \mathbf{W}_{33} \]

Thread

Thread

Thread

\[ \mathbf{W} : \text{Window} \]
5. System Features

5.1 Multiple Synchronized Windows/Monitors

With this feature, users can use VR,U2C to open and play a QTVR panorama movie across multiple frames/windows with the same or different viewing angles. They can also delay a multi-display system for viewing multiple screensized windows on separate monitors. However, for best appearance, they should use monitors with same screen size, frequency, resolution, color characteristics, etc. If desired, they could view many aspects of movie scenes on many monitors arranged around them, which might be called “panoramic panoramas.”

Users can set a millon width value for placing multiple windows contiguously, or modeling the frames of the monitors as millon-like borders. However, this value must be given in resolution size units (pixels), so that it can be used for wide view, millon wide view, and pan angle calculation. Fig. 7 shows how multiple synchronized windows of a panorama can be arrayed. The horizontal gap between them is the millon width.

5.2 Stereo Panorama Viewer

Stereographic viewing uses the multiple synchronized windows feature, but requires some modification on node ID and the movie itself. To view the stereo QTVR movies, users can choose one of three viewing techniques: Parallel, Cross-eyed, or Over/under viewing techniques. If the Parallel option is chosen, an opened movie pair will be displayed side-by-side with the movie for the left eye on the left, and the one for the right eye on the right. If the Cross-eye option is chosen, an opened movie pair will be displayed side-by-side as well, but with the movie for the left eye on the right, and the one for the right eye on the left. Finally if users want to use a special viewer called “Leavision,”15 they can select the Over/under option, so the stereo movie will be displayed vertically, with the movie for the right eye above that for the left.

To see a movie stereoscopically, the left eye must view the movie taken by the left camera, and the right eye must view the right movie. Using parallel and cross-eyed viewing techniques, they can be viewed without any form of viewing aid (free-viewing) [McA93], but the fusion takes a little practice because it involves paralleling or crossing one’s eyes slightly. However eye strain can be reduced by using a stereo viewer with a lens for each eye and special mounts like ScreenScope16 (Fig. 8) for viewing the movies. Alternatively, binocular HMDs can present stereographic pairs to individuals.

For projection method (Fig. 9), users can configure a dual display system sending the video signals for the left and right movies to separate projectors. These two projectors project through two out-of-phase polarized filters onto the same place on a silver (not white) screen with retroreflective properties that preserve polarization. Then users can enjoy the stereo effect with passive polarized eyewear.

---

Table 2: Shared Methods from Server

<table>
<thead>
<tr>
<th>Method</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>getOrientation()</td>
<td>To get/set roll, pitch and yaw values from/through server</td>
</tr>
<tr>
<td>setOrientation()</td>
<td></td>
</tr>
<tr>
<td>getExtraParam()</td>
<td>To get/set extra parameter values from/through server, such as ‘zoom’ value</td>
</tr>
<tr>
<td>setExtraParam()</td>
<td></td>
</tr>
</tbody>
</table>
6. Restrictions and Future Research

When playing back a QTVR movie, we cannot make the displayed image rotated to the left or right. Therefore this is an obvious limitation: no roll. Moreover the stereo stitching is not entirely seamless; there are subtle but distinct “hick-coughs” when switching between the multiple panorama movies used for one side of a stereo pair.

One of our motivating goals has been the idea of giving pictures depth, like that exploited by the “Esper” in the movie “Blade Runner,” which could extract almost limitless information from a single “hyper-still,” allowing users to look around corners and behind walls, seeing previously occluded objects. Such photographic omniscience (if not omnipotence) recalls the successive magnifications used in Michaelangelo Antonioni’s “Blowup.” “Deep pans” captures the idea of aligned multinode movies, which interpolate between the 2D of cylindrical/spherical geometry of QTVR and the 3D geometry of CAD. We can call this “multinode movies with smart zooming feature,” in which the system ndges the azimuth, elevation and zoom to align with panos captured from displaced viewpoints.

Other features for future research include:

- vertical viewing mode (which uses transoms instead of mullions to stack windows or monitors vertically),
- multifocal fisheye function (to allow inspection of portions of a panoramic scene at higher magnification),
- dynamic window (contraction/dilation) sizing,
- subliminal scenes (rapidly switching panoramic movies),
- object movies,
- spatial sound (delivered via authoring tools like SoundSaVR or directly at runtime), and
- integration with other clients.

\[17 \text{www.mgyr.com}\]
7. Conclusion
The VR$U_2$C multi-monitor and -display QTVR browser, integrated with our heterogeneous groupware client suite, enables multimodal activity. Although stereo panoramic viewers have been developed, as well as multimonitor applications, to the best of our knowledge, ours is the first instance of non-anaglyphic stereoscopic QTVR.

References


