VR_4U_2C: QuickTime と Java による
マルチモニタとステレオグラフィックに対応した
マルチユーザ, マルチパースペクティブ パノラマブラウザ
A Multiuser Multiperspective Panoramic Browser Using QuickTime VR and Java Featuring Multimonitor and Stereographic Display

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. Although many hardware components for constructing such immersive visual displays may be purchased at reasonable cost, no general-purpose software system was readily available to integrate these devices for computer-based interactive control. To address these needs in particular for the more specialized components typically used in multiuser, multiperspective, panoramic and stereographic image display systems, we wave 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 practically only by the size and number of monitors or projectors assembled around individual or groups of users in various viewing locations. Named "VR_4U_2C" ("virtual reality for you to see"), the browser, developed platform-agnostically 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 and interoperates seamlessly with them. VR_4U_2C 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, rotary controller, track pad, and keyboard. Furthermore, VR_4U_2C provides a unique solution to the problem of interactive stereoscopic display of QTVR imagery.

Keywords: QuickTime VR (QTVR), Java, multiuser, multiperspective, multidisplay, panoramic scene.

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 (QTVR)[1] [21] technology. Whether the visual environment to be displayed is computer generated or photographically captured, the browser, named "VR_4U_2C" ("virtual reality for you to see") [10], can immerse a viewer or viewers in that environment using a wide variety of spatial configurations of visual display devices. Figure 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 VR_4U_2C 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. Unlike a surround screen projection system that provides natural stereoscopic depth cues for only one of several simultaneous viewers in a shared physical space (like [5], which approach uses processed continuous video streams as contrasted to our segues between discrete panoramic snapshots), our browser was designed to enable other viewers to enjoy the optimal stereo from their own vantage point in a separate physical site.

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 mosaic-based video wall [30], or more immersive systems such as the CAVE [14], an arbitrary number of displays with potentially diverse features has been envisioned. Rather than attempting the continuous coverage depicted in Figure 1, the flexibility of VR_4U_2C suggests the idea of "panoramic panoramas" that piecewise

[1] www.apple.com/quicktime/qtvr

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 dis-contiguous. Such a heterogeneous image-based approach to the presentation of and navigation through a virtual environment does not seem to have been anticipated in the initial development of QTVR [11], though the concept is certainly not without precedent (see, for example, [28]).[2]

This paper focuses upon only a few potential display-device configurations, in order to explain implementation details and functions of the browser. We begin with an introduction to basic concepts in spatialized hypermedia, including panoramic and stereographic display. This introduction necessarily includes a discussion of the number of degrees of freedom supported by an interactive image display system, a detail which is central to distinctions between the current system and more conventional systems.

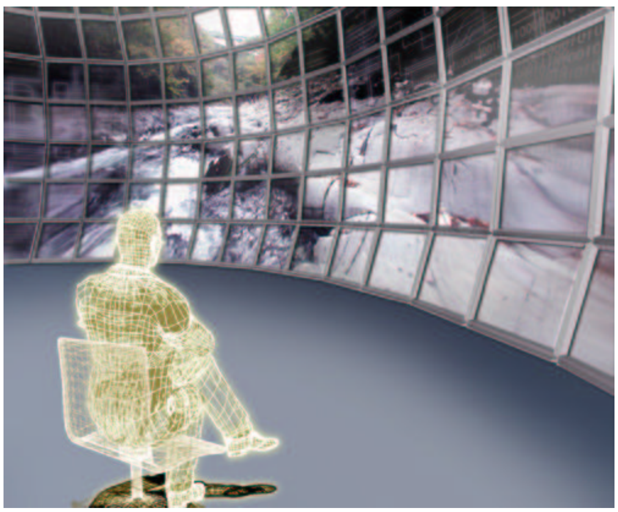


Figure 1: Graphic Depiction of a Curved "Video-Wall" Configuration of Visual Display Devices Showing Immersive QTVR Imagery to a Viewer. (Graphic produced by "Eyes, Japan."[4])

1.1 Hypermedia and Stereo Panoramas

Hypermedia combines the "hot links" characteristic of hypertext systems with media elements (or multimedia— including sound, graphics and images, audio, video, animation), making use of a variety of interactive tools to integrate them in a meaningful way. Inasmuch as hypermedia allows experiential display and manipulation of data, it can be thought of as equivalent to virtual reality, artificial reality, and cyberspace.

The word "panorama" was introduced in 1792 in London to describe a large cylindrical room entirely covered by a 360° painting [27] [9]. 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 QTVR, Helmut Dersch's PTViewer,[5] and iPIX Viewer[6]— can be produced by tiling multiple rectangular images or by combining images captured by a fish-eye lens.

Such systems typically 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). These terms are borrowed from the language used in controlling cameras [8] [31], but these details become more clear in the systematic description of the several dimensions of spatial motion that can be experienced in the "real world." These dimensions include translational degrees-of-freedom (sway, heave, and surge), and rotational degrees-of-freedom (roll, pitch, and yaw), as shown by Table 1.

In QTVR, only two rotational degrees-of-freedom are typically supported: the up–down pitch of the viewing angle and its left–right yaw. Therefore, a user request for a barrel roll, as an airplane offers, could not be granted. Although QTVR allows the viewer to "zoom in" on details of a scene, such control only seemingly supports viewer spatial motion. In fact, the zoom control determines only 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. If, instead of a "zoom in" on details, an interactive display allowed the viewer to "dolly in" on those details, then nearby objects would grow in visual angle at a rate greater than more distant objects, a characteristic very rare in panoramic image display, though not so rare in image-based rendering (see, for example, [29]). It should be clear that the relative angular subtense of imaged objects changes for dolly but not for zoom. When viewing a scene with the unaided eye, the experience of zooming never occurs; rather, moving towards an object causes the object to "loom" rather than to "zoom." Even when few distinct objects are present in a scene, such as a panoramic view of a field of grass, the nearby textures flow in a manner that cues viewer motion (a phenomenon termed "optical flow" by Gibson [16].

In descriptions of such phenomena, readers may be more familiar with the term "motion parallax" as an important cue to visual depth. Of course, the clas-

[2] A notable artistic example of immersive panoramic image display was the "Placeholder" VR piece created by Laurel, Strickland, and Tow [22], which was influential on the development of the system described in this paper due to their interesting use of somewhat "patchy" coverage of virtual landscapes (rendered by "panoramic tiling of multiple video images onto a spherical wireframe").

[4] www.aizu.com

[5] www.fh-furtwangen.de/~dersch/PTVJ/doc.html

[6] www.ipix.com

Position					
Static		**Dynamic**			
Location	**Scalar**	**Translation; 平行移動 (へいこういどう)**		**Along Axis**	**Perpendicular to Plane**
lateral displacement	abscissa x	sway; 左右 (さゆう)	left↔right	x	sagittal (median)
frontal displacement	ordinate y	surge; 前後 (ぜんご)	back (aft): retreat ↖↘ forth (fore): advance	y	frontal
height	altitude z	heave; 上下 (じょうげ)	up: ascend ↕ down: descend	z	horizontal
Orientation or Attitude		**Rotation; 回転 (かいてん)**		**About Axis**	**In Plane**
elevation or tilt	ϕ	pitch (tumble, flip); 縦方向 (たてほうこう)	climb/dive	x	sagittal (median)
(roll)	ψ	roll (flop); 横方向 (よこ)	left/right	y	frontal
azimuth	θ	yaw (whirl, twist); 偏方向 (かた or へん)	CW/CCW	z	horizontal

Table 1: Physically spatial dimensions: taxonomy of positional degrees of freedom.

sic demonstration of a motion parallax effect is not to stage a back–forth surge, but rather a left–right sway. That is, translational motion of the viewer (stepping to the side) is used to reveal an object or scene element that was hidden behind another object or scene element, as illustrated by the x-axis in Figure 2; but movement away from a foreground object, along the z-axis ("Depth") shown in Figure 2, can also reveal previously hidden background objects. In deeper investigation of such phenomena, the relation to visual depth between such motion-dependent monocular cues and the more temporally immediate cue termed "binocular parallax" needs to be considered. Though a proper review of this interesting area of vision science is well beyond the scope of this paper, some additional discussion of these two strong cues to visual depth are to be examined in this introduction, with the goal of clarifying the differences between various interactive panoramic display systems, and revealing the novelty of the VR_4U_2C browser system we have developed.

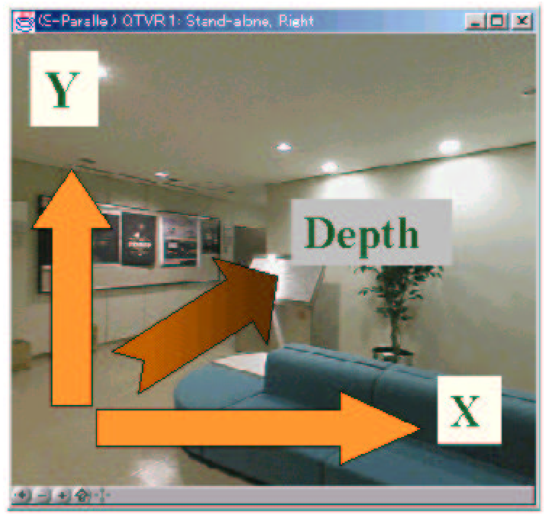


Figure 2: Depth: 3^{rd} Dimension of Stereo Images

First, it should be pointed out that panoramic stereoscopic display is quite rare, and is typically only attempted in expensive, specialized systems such as the CAVE-based system [14] described by [5]. Consideration of stereoscopic imagery for long-range views, such as panoramic landscape imagery, is less important than for mid-range views [18], but the user's impression given panoramic imagery of everyday indoor environments will almost always be enhanced by stereoscopic viewing. Whereas an ordinary stereo display presents only a single pair of images, one to each eye, a stereo panorama requires more than two points of view [28]. In the system described in this paper, we use multiple panorama movies of the same scene captured from slightly displaced points of view (separated by an interocular distance). Then, using one of many "binocular image selection" technologies, the displayed stereoscopic movies can be fused in the "cyclopean eye" to create a solid-seeming, third dimension of depth in visualizing panoramic scenes of spaces using several single windows on the space. Other systems have been designed to create panoramic stereoscopic imagery from movies using a single video camera [4], which offers some advantages over the system described in this paper. Capturing a sequence of images using a novel "point card" gives the current system some unique features that are described in subsequent sections below. First, however, the aims and objectives of the system are defined in the following section.

2 Aims and Objectives

This research involved development of a multiuser multiperspective panoramic browser using Apple's QuickTime VR technology and the Java programming language.

The main objectives of the research are:

- To clearly identify and understand contemporary methods for capturing mono and stereo panoramic scenes, producing QTVR movies.
- To develop a Java program that can be used to interactively explore QTVR movies with other users in realtime through the network.
- To integrate QTVR movies seamlessly with other clients in the Spatial Media Group's Multimodal Groupware suite [20], including our Internet Chair [6] [12], 2.5D Dynamic Map [25], Pioneer Sound Field Controller [2], RSS-10 Speaker Array Driver [1], Java3D widgets [3], Soundscape Stabilized (Swivel Seat) Spiral Spring [13], and DoCoMo iαppli "ι·Con" application for mobile phones [26].
- To elaborate more interesting features for viewing QTVR movies on computer and other equipment such as TV screens, head-mounted displays (HMDs), stereo projection systems, etc.

3 Related Systems

Many advanced panoramic viewers have been developed recently, including the well-known Elumens[7] VisionDome and VisionStation. The VisionDome can deliver a full-color, high-resolution, raster-based, interactive, 3D display, with a $360° \times 180°$ panoramic projection on the interior of a hemispherical dome. A dozen or more people can simultaneously collaborate without having to use restrictive head-mounted displays. 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.

VisionStations are more affordable, lower-maintenance and easier to use than VisionDomes. With a ceiling height of 8 feet and audience capacity of one–six people, the VisionStation is appropriate for locations with small work groups or audience sizes. The standard VisionStation can be used for reviewing buildings, automotive and aerospace designs, or as a low cost driving simulator.

[7] `www.elumens.com`

4 Implementation Details

4.1 Monocular Panoramic Capture

Panoramic scenes can be photographically captured using a Nikon[8] CoolPix 990 digital camera with its MC-EU1 remote shutter release, CompactFlash[9] memory card, card reader, Kaidan[10] and EyeSee360[11] 360 One VR optical system with CoolPix 990 mounting kit, and a standard monopod with bubble level. With a single 3.34 megapixel CoolPix 990 shot, the 360 One VR mirrored optical system can provide a complete 360° panorama with a 100° vertical field-of-view (50° above and 50° below the horizon). PhotoWarp[12] software can be used to process the captured panoramic image (as shown in Figure 3), yielding a QuickTime movie or cylindrical image.

4.2 Stereo Panoramic Capture

The equipment used for capturing and preparing stereo panoramic scenes includes all that for ordinary panoramic capture plus a compass and a "point card" as shown in Figure 4. Firstly, place a monopod at the center of the card (`point 1`) and take a picture, capturing a complete 360° panoramic scene. As for ordinary panoramic capture, this panoramic image is processed and then saved as a QuickTime movie, designatable as a left-eye panorama.

To make the right-eyed side of a panoramic scene, take another picture after displacing the monopod an interocular distance (nominally 65 mm, to `point 2`), adjusting so that the compass attached to the monopod indicates the same direction as the capture for the left-eyed panorama, ensuring that every captured panoramic scene will align with as the original panorama. Repeat until all necessary pictures have been captured, and compile these six complete 360° panoramic images into a multinode QuickTime movie, which can be designated the right-eye panorama.

To handle this multinode stereographic feature, consider the numbers for each QTVR movie node as shown in Figure 4. Upon opening a stereographic QTVR movie with an initial pan angle (0°), VR_4U_2C will display `node 1` as the left-eye panorama and `node 2` as the right-eye panorama, as shown in Figure 5. While panning to the left (increasing pan angle) or to the right (decreasing pan angle), VR_4U_2C will monitor the pan angle value and constantly use `node 1` as the node for the left-eyed view. However, if the pan angle reaches a certain threshold,

[8] `www.nikon.com`
[9] `www.compactflash.org/info/cfinfo.htm`
[10] `www.kaidan.com`
[11] `www.eyesee360.com`
[12] `www.eyesee360.com/photowarp`

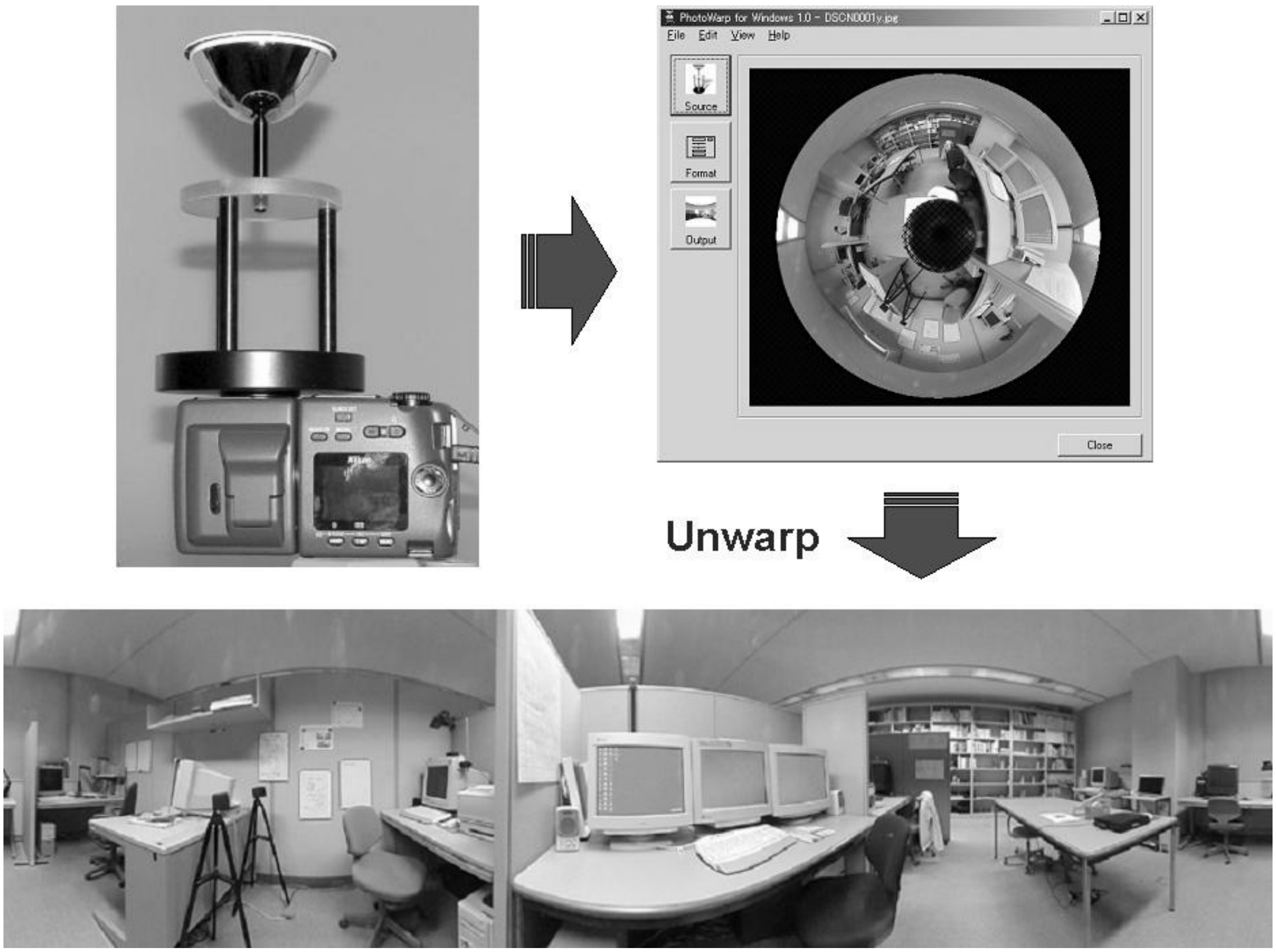


Figure 3: Preparing Panoramic Scene

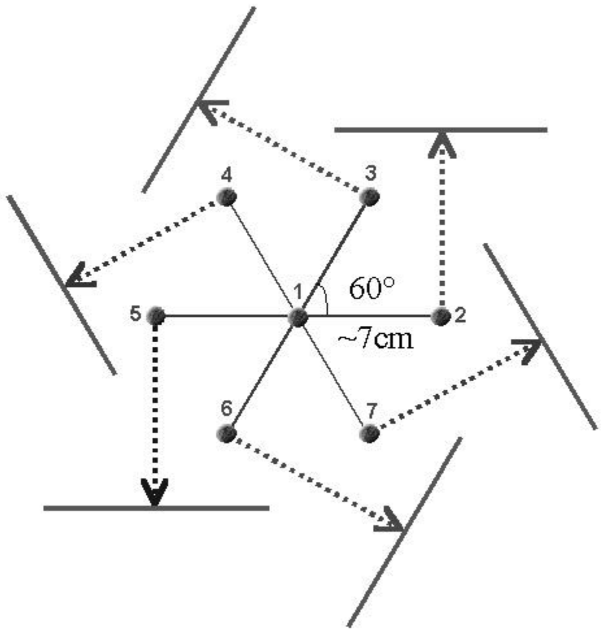


Figure 4: Capturing Stereo Panoramas: discrete sampling using a point card with nodes numbered 1–7

Figure 5: Example of Left/Right Stereo Images

VR_4U_2C will change the displayed node for the right-eyed panorama to the appropriate node.

In this case, a revolution about a point (left-eye point) occurs and both orientation and location change. Unlike an ordinary panorama that only rotates on a single point changing orientation (yaw), the multinode stereographic feature allows objects to loom, relative angle subtense changing for dolly but not for zoom. 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 stero pair. Although this looming effect might be regarded as an unwanted artifact of our implementation, fixing the position of one eye's view whilst revolving the other, the improved depth impression that comes from the slight sway and surge of the observer is a definite advantage of this implementation. A method using a higher spatial-sampling rate (more capture-intensive) would avoid such welcome side-effects, but that slight sway and surge that comes virtually for free using more coarse sampling would be lost. As an alternative, rotation about the center point of the interocular axis could provide smoother rotation with less obvious sway and surge, since both eye points symmetrically rotating about a fixed midpoint would complete one cycle of sway and surge with every 360° rotation of the viewer.

4.3 Groupware Architecture

Figure 6 illustrates the relationship between VR_4U_2C and other clients in our Spatial Media Group's Multimodal Groupware suite [7]. As one of many integrated clients, VR_4U_2C 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 in `CVEClient` class, an instance of which is linked to our application.

Method	Event
`getOrientation()` `setOrientation()`	To get/set roll, pitch, and yaw values from/through server
`getExtraParam()` `setExtraParam()`	To get/set extra parameter values from/through server, such as "zoom" and "node" values

Table 2: Shared Methods from Server

Upon receiving values from a server using the `getOrientation()` method, VR_4U_2C will assign pitch to tilt angle value, and yaw to pan angle value.

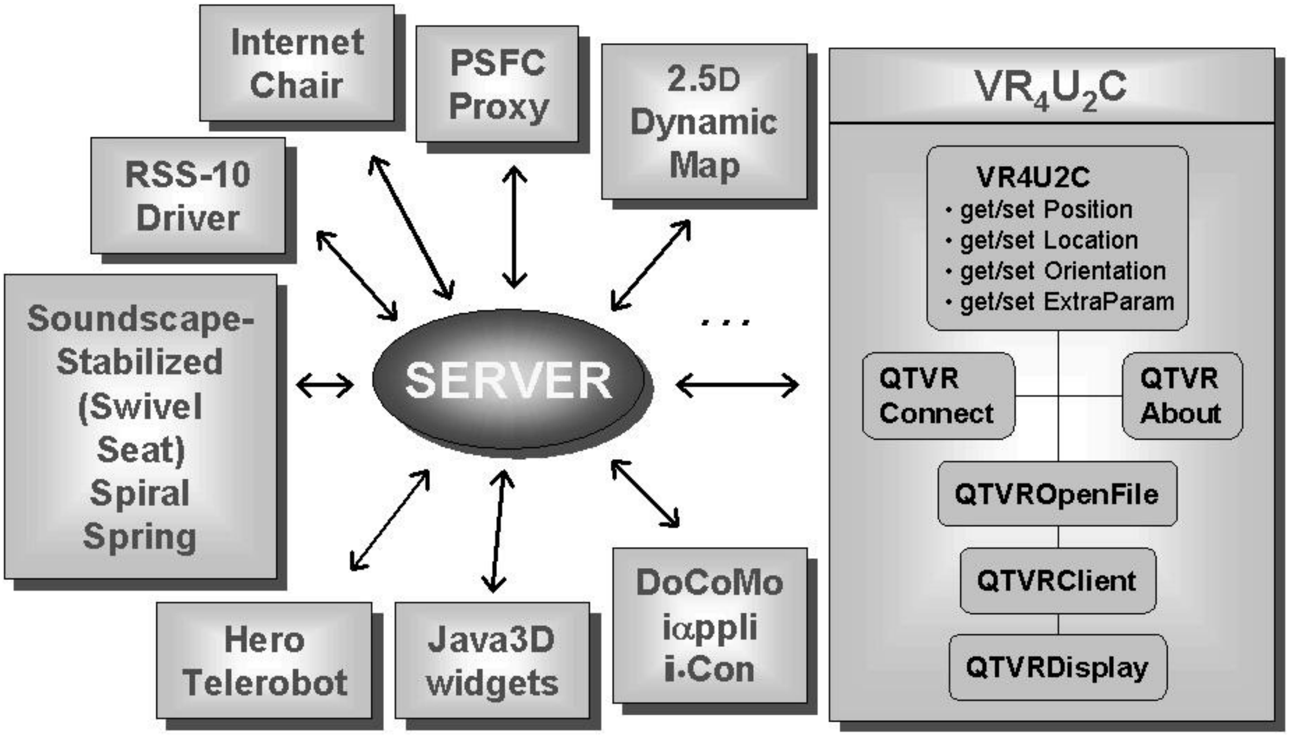


Figure 6: System Structure

However, the program will only cache the roll value. (Our system has no provision for making a displayed image roll.) If the user changes tilt or pan angle, the new state will be multicast through the server using the `setOrientation()` method. VR_4U_2C uses the `getExtraParam()` method to get "zoom" and "node" values from the server, assigning them to the field-of-view and the current node data-fields. Symetrically, if these variables are updated, they will be sent to the server using `setExtraParam()` method.

4.4 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 [23]. These callbacks are used in VR_4U_2C to perform tasks when certain conditions arise within QuickTime itself. The `MovieDrawingComplete` callback is used to notify a supervising program whenever QuickTime has drawn to the screen. Once a movie draws on the screen, an instance of the subclass (`MovieDrawing`) implementing this callback will be called automatically, and will compute view width (W), mullion width (M), and pan angle (PA) of all the associated windows, as shown in Figure 7.

Furthermore, there are two more QTVR callbacks provided in QuickTime VR: the `QTVRInterceptor` callback for panning, tilting and zooming processes, and the `QTVREnteringNode` callback for entering node processes. The subclass implementing the `QTVRInterceptor` callback contains a number of useful methods, outlined in Table 3.

Method	Event
`getPanAngle()` `setPanAngle()`	To get/set current pan angle
`getTiltAngle()` `setTiltAngle()`	To get/set current tilt angle
`getFieldOfView()` `setFieldOfView()`	To get/set current field of view
`ptToPanAngle()`	To get pan angle at certain point (in pixels) on movie window
`goToNodeID()`	To move to another node in multinode movie

Table 3: Relevant Callback Methods

4.5 Multithreaded Implementation

The benefits of multithreading are better interactive responsiveness and realtime behavior [17, p. 10]. Upon opening a movie, VR_4U_2C will create a thread for its own operation, and other threads for displaying each window/frame, as shown in Figure 8. Every process in the opened movies can be run in parallel on the same machine.

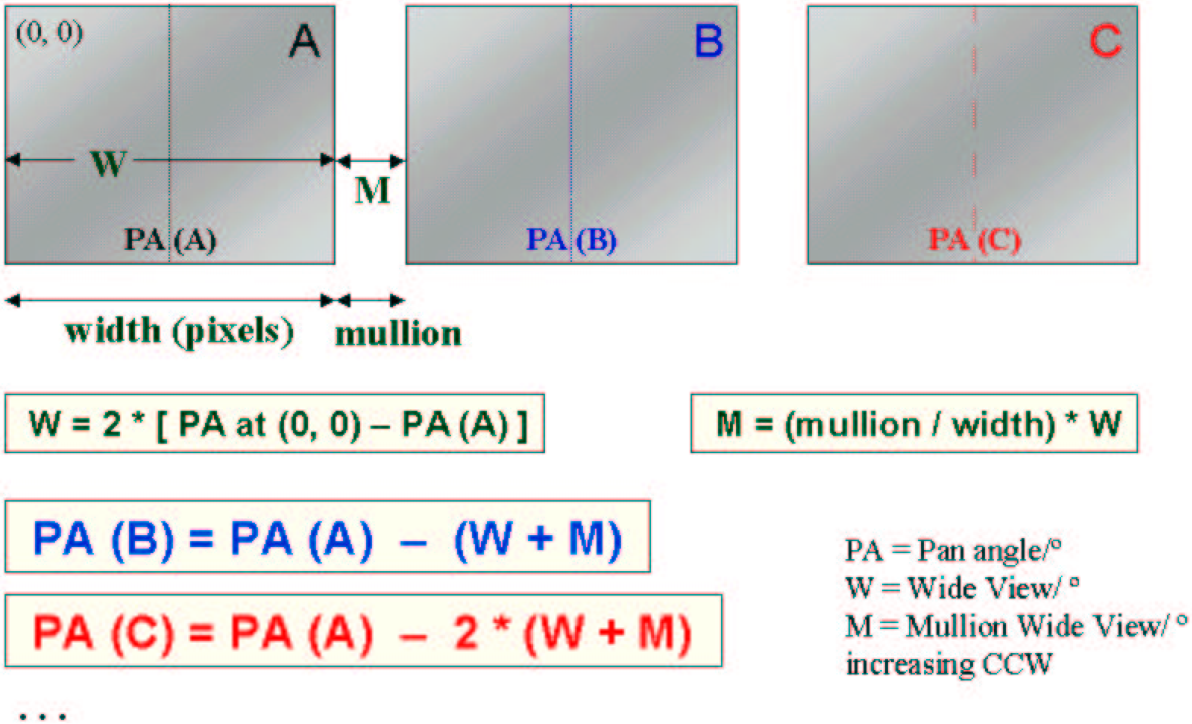


Figure 7: Calculations for View Width, Mullion Width, and Pan Angle

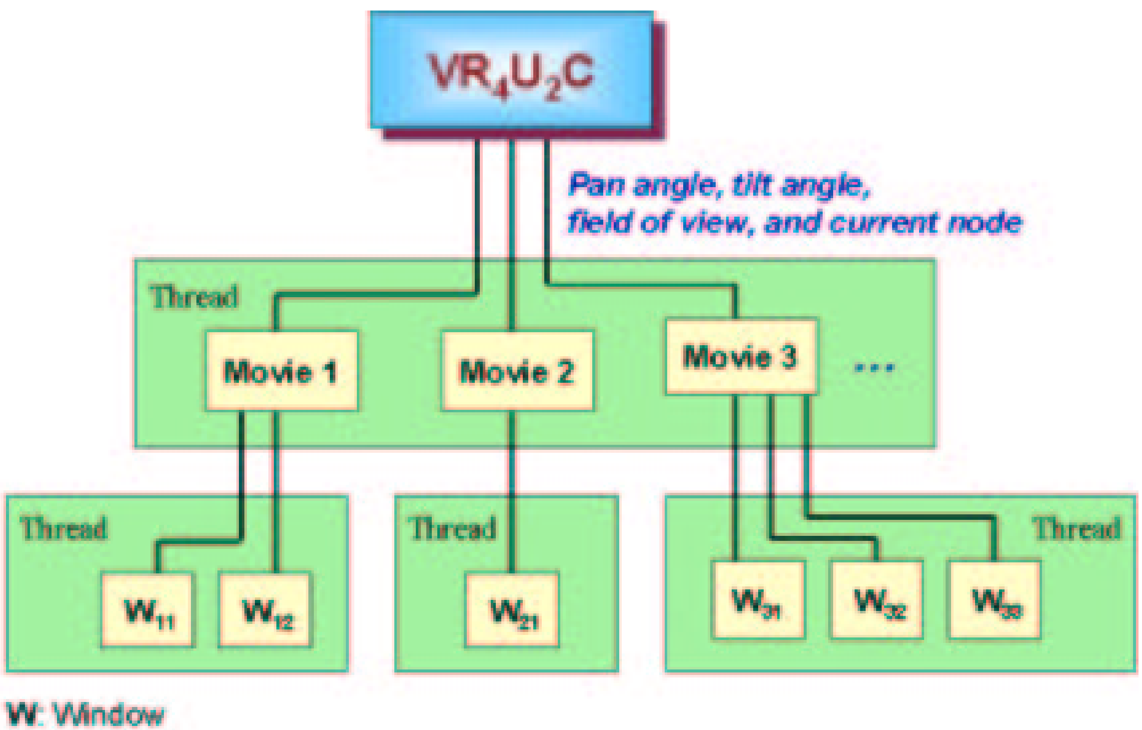


Figure 8: Example of Multithreaded Execution

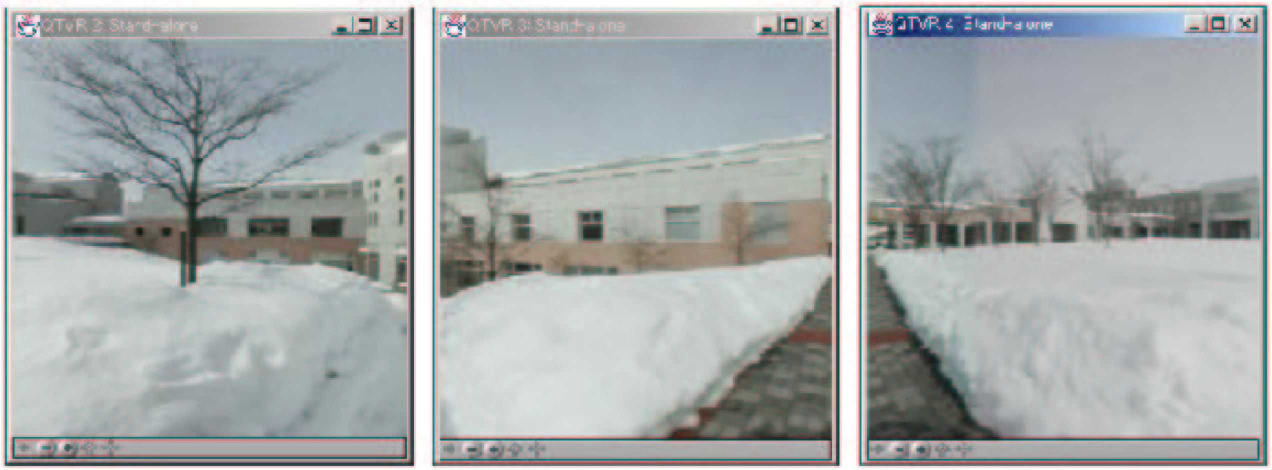


Figure 9: Multiple Synchronized Windows

5 System Features

5.1 Multiple Synchronized Windows/Monitors

With the multiple synchronized displays feature, users can use VR_4U_2C to open and play a QTVR panorama movie across multiple frames/windows with the same or different viewing angles. They can also deploy a multidisplay system for viewing multiple screen-sized windows on separate monitors. (Naturally monitors with consistent screen size, frequency, resolution, color characteristics, etc. should be used for best appearance.) If desired, one could view many aspects of movie scenes on many monitors arranged around them, which arrangement might be called "panoramic panoramas." Users can set a mullion width value for placing multiple windows contiguously, or modeling the frames of the monitors as mullion-like borders. However, this value must be given in resolution size units (pixels), so that it can be used for view width, mullion width, and pan angle calculations. Figure 9 illustrates how multiple synchronized windows of a panorama can be arrayed. The horizontal gap between them is the mullion width.

5.2 Stereo Panorama Viewer

Stereographic viewing uses the multiple synchronized windows feature, but requires some elaboration on selection of node ID and the movie itself. To view stereo QTVR movies, users can choose one of three viewing techniques: `parallel`, `cross-eyed`, or `over/under`. 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 stereo pair swapped. Finally if users want to use a special viewer called "Leavision,"[13] the `over/under` option will arrange the stereo pair vertically, with the movie for the right eye above that for the left.

To see a panorama 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, the pair can be viewed without any form of viewing aid ("free-viewing") [24], but such 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 like the Berezin ScreenScope[14] (Figure 10), which contains adjustable mirrors that reflect a stereo-image pair from the computer screen onto one's eyes, behind which the brain fuses the images into a single, sharp 3D scene. Since it uses front-surface mirrors, the ScreenScope supports full color, isn't haunted by ghost images, is completely free from screen flicker, which allow for easy viewing of many sizes of images. The monitor's resolution determines the 3D image resolution. Alternatively, binocular HMDs like iO Display Systems' i-glasses[15] can be used to present stereographic pairs to individuals.

For projection method (Figure 11), users can configure a dual display system sending the video signals for the left and right movies to separate projectors. These two projectors shine through two out-of-phase polarized filters onto the same place on a silver (not white) screen (like that sold by Reel 3-D Enterprises[16]) with retroreflective properties that preserve polarization, allowing users to enjoy stereo effects with passive polarized eyewear.

[13] `www.stereoscopy.com/faq/waack-ch-5.html`
[14] `www.berezin.com/3d/screenscope.htm`
[15] `www.i-glassesstore.com/hmds.html`
[16] `www.stereoscopy.com/reel3d/screens.htm`

Figure 10: Viewing Side-by-side Stereo Images using ScreenScope Desktop, Laptop, and Handheld. (Photos courtesy of Berezin Stereo Photography Products, USA.)

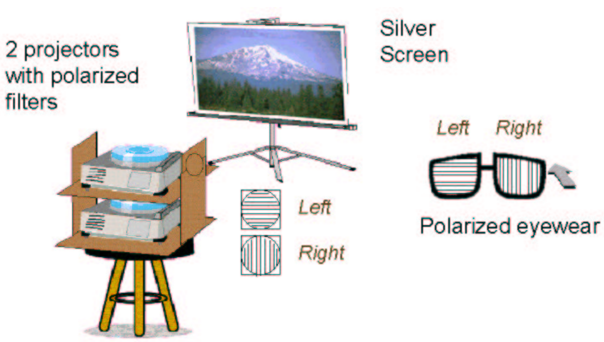


Figure 11: Screen Viewing Using Polarized Projection Filters and Eyewear

6 Future Research and Conclusion

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 recalls the successive magnifications used in Michaelangelo Antonioni's "Blowup." "Deep panos" captures the idea of aligned multinode movies, which interpolate between the 2D cylindrical/spherical geometry of QTVR and the 3D geometry of CAD. We are considering developing such a "multinode movies with smart zooming" feature, in which the interface uses "snap-to" features to nudge the azimuth, elevation, and zoom to align with panos captured from displaced viewpoints.

Other extensions for future research include:

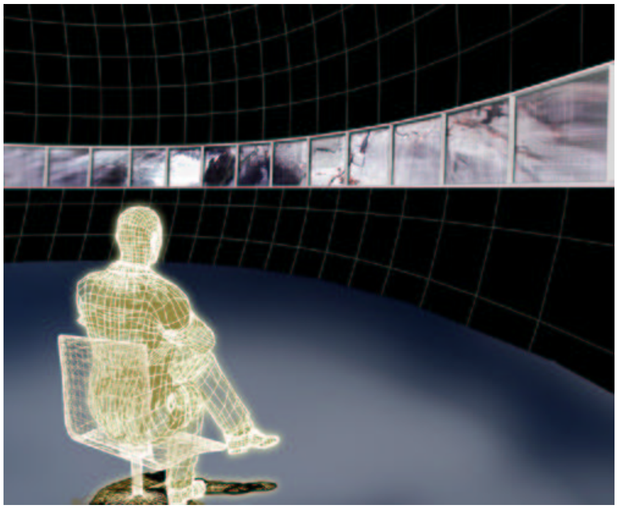
- vertical (which uses transoms instead of mullions to stack windows or monitors vertically) or pyramidal, etc. tiling arrangements[17] (besides already-implemented horizontal tiling), like those suggested by Figure 12,
- panoramic panorama, as described in the Introduction, arranging coordinated immersive displays around users,
- multifocal fisheye function [15] (to allow inspection of portions of a panoramic scene at higher magnification),
- dynamic window (contraction/dilation) sizing,
- subliminal scenes (rapidly switching panoramic movies), and
- object movies.

The VR_4U_2C 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 stereographic QTVR.

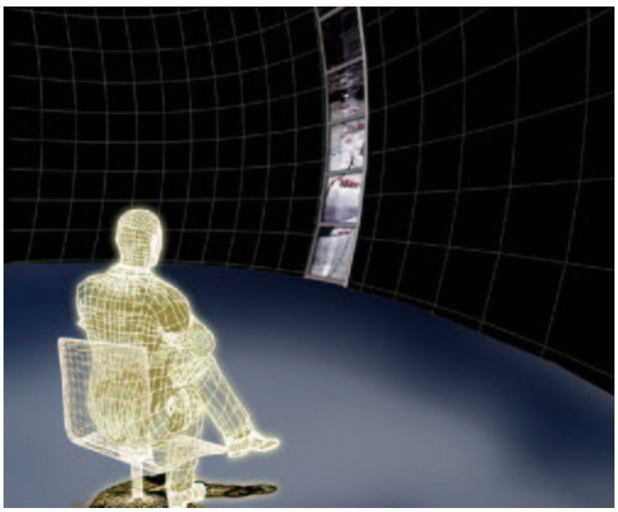
Acknowledgement

Helpful suggestions from anonymous referees contributed significantly to the improvement of this paper, and we offer them our thanks.

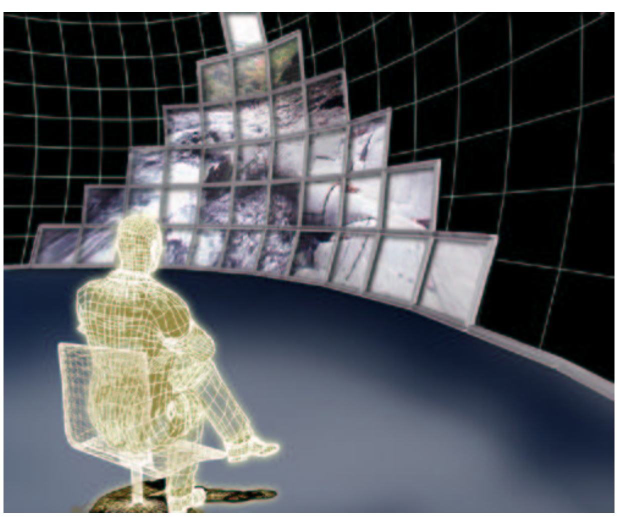
[17] www.wrightline.com/productDetail.asp?ProductID=14&ProductCategoryID=9&SubCategoryID=0



(a) Horizontal



(b) Vertical



(c) Pyramidal

Figure 12: Three Tiling Styles. (Graphics by「あいづ・ジャパン」.)

References

[1] 公園 マイケル, 安積 卓也, 八柳 芳生, 佐々木 正浩, 山岡 想, and 武市 修. ネットワークと連携するスピーカ群に生成される音像のクライアントへの転送手段とされるストリーミングの実装：世界で最も高価な立体音響システムの構築. In *ICAT: Proc. **I**nt. **C**onf. on **A**rtificial Reality and **T**ele-Existence*, pages 162–169, 東京大学, 12 月 2002.

[2] 天野 克巳, 松下 文雄, 柳川 博文, 公園 マイケル, イエンツ ヘルダー, ウイリアム L. マーテンス, 木場 美治, and 東山 三樹夫. マルチスピーカ方式による２音源立体音像制御— A Virtual Reality Sound System Using Room-Related Transfer Functions Delivered Through a Multispeaker Array: The PSFC at the University of Aizu Multimedia Center. *TVRSJ: 日本バーチャルリアリティ学会論文誌 (**T**rans. of the **V**irtual **R**eality **S**ociety of **J**apan)*, 3(1):1–12, 3 月 1998. ISSN 1344-011X.

[3] ノア アラムシャ ボルハサン, 公園 マイケル, ニュトン ファナンド, 鎌田 智也, ウイリアム L. マーテン ン, 尾坂 弘章, and 吉川 拓蔵. 『自分自身をご覧なさい』：立体視システムにおける自己中心的な視点移動の外側からの可視化とエミュレーション. In *Proc. ICAT: **I**nt. **C**onf. on **A**rtificial Reality and **T**ele-Existence*, pages 146–153, 東京大学, 12 月 2002.

[4] 川北 泰宏, 濱口 佳孝, 塚本 明利, and 宮崎 敏彦. 単眼動画像からのパノラマステレオ画像の生成. **情報処理学会論文誌**, 41(6):1848–1856, June 2000.

[5] 苗村 健, 金子 正秀, and 原島 博. 大型スクリーン環境における多人数鑑賞用ステレオ表示. In **日本バーチャルリアリティ学会大会論文集**, pages 57–60, 8 月 1998.

[6] 小泉 宣夫, コーエン マイケル, and 青木 茂明. 特許 #3042731: 音声再生装置, 3 月 2000. `www6.ipdl.jpo.go.jp/Tokujitu/PAJlayout.ipdl?N0000=62&N0120=01&N2001=2&N%3001=H05-037994`.

[7] 公園 マイケル and ウイリアム L. マーテンス. 会津大学のスペーシャル メディア. **日本バーチャルリアリティ学会誌** *(JVRSJ: **J**. of the **V**irtual **R**eality **S**ociety of **J**apan)*, 6(2):52–57, 9 月 2001. ISSN 1342-6680; `www.u-aizu.ac.jp/~mcohen/welcome/publications/spatial-media.ps`.

[8] Daniel Arijon. *Grammar of the Film Language*. Silman-James Press, 1991. ISBN 1-87950-07-X.

[9] Ryad Benosman and Sing Bing Kang. *Panoramic Vision: Sensors, Theory, and Applications*. Springer, 2001. ISBN 0-387-95111-3.

[10] Noor Alamshah Bolhassan, William L. Martens, and Michael Cohen. VR_4U_2C: A Multiuser Multiperspective Panoramic Browser using QuickTime VR and Java Featuring Multimonitor and Stereographic Display. In *Proc. ICAT2001:* 11th *Int. Conf. on* **A***rtificial Reality and* **T***elexistence*, pages 161–168, Tokyo, Japan, December 2001. ISSN 1345-1278.

[11] S. E. Chen. QuickTime VR— An image-based approach to virtual environment navigation. In *Computer Graphics (Proc.* SIGGRAPH *'95)*, pages 29–38, 1995.

[12] Michael Cohen. The Internet Chair. *IJHCI: Int. J. of Human-Computer Interaction*, 15(2), 2003. `www.u-aizu.ac.jp/~mcohen/welcome/publications/ic4.ps`.

[13] Michael Cohen and Kenta Sasa. An Interface for a **S**oundscape-**S**tabilized **S**piral-**S**pring **S**wivel-**S**eat. In *Proc. WESTPRAC VII:* 7th **West***ern* **P***acific* **R***egional* **A***coustics* **C***onf.*, pages 321–324, Kumamoto, Japan, October 2000. ISBN 4-9980886-1-0 and 4-9980886-3-7.

[14] C. Cruz-Neira, D. Sandin, and T. DeFanti. Surround-Screen Projection-Based Virtual Reality: Design and Implementation of the CAVE. In *Computer Graphics (Proc.* SIGGRAPH *'93)*, pages 135–142, 1993.

[15] George W. Furnas. Generalized fisheye views. In *Proc. CHI:* ACM *Conf. on* **C***omputer-***H***uman* **I***nteraction*, pages 16–23, Boston, April 1986.

[16] J. J. Gibson. *The ecological approach to visual perception.* Houghton Mifflin, Boston, 1979. ISBN 0-89859-959-8.

[17] Cay S. Horstmann and Gary Cornell. *Core Java 2, Volume I–Fundamentals.* Prentice Hall, 2001. ISBN 0-13-089468-0.

[18] T. Ichikawa, S. Iwasawa, K. Yamada, T. Kanamaru, T. Naemura, K. Aizawa, S. Morishima, and T. Saito. *Three Dimensional Image Capturing and Representation for Multimedia Ambiance Communciation*, chapter 10, pages 227–240. In Javidi and Okano [19], 2002. ISBN 3-540-43549-2.

[19] Bahram Javidi and Fumio Okano, editors. *Three Dimensional Television, Video, and Display Technologies.* Springer-Verlag, 2002. ISBN 3-540-43549-2.

[20] Toshifumi Kanno, Michael Cohen, Yutaka Nagashima, and Tomohisa Hoshino. Mobile Control of Multimodal Groupware in a Distributed Virtual Environment. In *Proc. ICAT2001:* 11th *Int. Conf. on* **A***rtificial Reality and* **T***elexistence*, pages 147–154, Tokyo, Japan, December 2001. ISSN 1345-1278.

[21] Susan A. Kitchens. *The QuickTime* VR *Book: Creating Immersive Imaging on Your Desktop.* Peachpit Press, 1998. ISBN 0-201-69684-3.

[22] Brenda Laurel, Rachel Strickland, and Rob Tow. Placeholder: Landscape and narrative in virtual enviroments. *ACM Computer Graphics Quarterly*, 28(2), May 1994. `www.tauzero.com/Brenda_Laurel/Severed_Heads/CGQ_Placeholder.html`.

[23] Tom Maremaa and William Stewart. *QuickTime for Java: A Developer Reference.* Morgan Kaufman: Academic Press, 1999. ISBN 0-12-305440-0.

[24] David F. McAllister, editor. *Stereo Computer Graphics and Other True 3D Technologies.* Princeton University Press, 1993. ISBN 0-691-08741-5.

[25] Takashi Mikuriya, Masataka Shimizu, and Michael Cohen. A Collaborative Virtual Environment Featuring Multimodal Information Controlled by a Dynamic Map. *3D Forum: J. of Three Dimensional Images*, 15(1):133–136, 3 2001. ISSN 1342-2189.

[26] Yutaka Nagashima and Michael Cohen. Distributed virtual environment interface for a mobile phone. *3D Forum: J. of Three Dimensional Images*, 15(4):102–106, December 2001. ISSN 1342-2189.

[27] Michael Naimark. Elements of realspace imaging: A proposed taxonomy. In *Proc. SPIE/SPSE Electronic Imaging Conf.*, San Jose, CA, 1991. Vol. 1457.

[28] Michael Naimark. A 3D Moviemap and a 3D Panorama. In *Proc. SPIE/SPSE Electronic Imaging Conf.*, San Jose, CA, 1997. Vol. 3012.

[29] Jonathan W. Shade, Steven J. Gortler, Li-Wei He, and Richard Szeliski. Layered Depth Images. In *Computer Graphics (Proc.* SIGGRAPH *'98)*, pages 231–242, Orlando, Florida, July 1998.

[30] R. Szeliski and H.-Y. Shum. Creating full view panoramic image mosaics and environment maps. In *Computer Graphics (Proc.* SIGGRAPH *'97)*, pages 251–258, 1997.

[31] Jeremey Vineyard. *Setting Up Your Shots: great camera moves every filmaker should know.* Michael Wiese Productions, 2000. ISBN 0-941188-73-6.

(2002 年 7 月 1 日受付)