

3-5 gCubik+i Virtual 3D Aquarium: Interfacing a Graspable 3D Display with a Tabletop Display

Roberto Lopez-Gulliver, YOSHIDA Shunsuke, MAKINO Mao,
YANO Sumio, and ANDO Hiroshi

We have proposed, and developed, the “gCubik”, a cubic-shaped, graspable and glasses-free 3D display, as a communication tool to support group collaboration. The display provides multiple users with correct viewpoint-dependent full parallax 3D images.

This paper presents the current gCubik’s 6-face prototype and its design concept. First, details of the following technical contributions are presented: a) wide field-of-view Integral Photography; b) real-time rendering for integral photography; and c) real-time animation support architecture for application development. Finally, we proposed the gCubik+i as a new interactive platform that naturally interfaces a 3D display with a tabletop display. A virtual 3D aquarium application that showcases the proposed platform’s capabilities and its interactivity potential is presented.

Keywords

Tabletop interaction, 3D display, Auto-stereoscopic display, Integral Photography, gCubik

1 Introduction

In order to realize natural human-to-human communication, basic research on transmission of multi-sensory data, including sight, smell, touch and sound, is necessary[1]. In the case of face-to-face collaborative tasks, users often rely on manipulating physical objects on top of a table in order to support their discussion of ideas. Computer-supported collaborative work enhances users’ discussions by adding interactivity to the table’s surface and introducing tangible virtual objects, using small handheld input/output devices. A natural interface between the tabletop display and the tangible virtual objects is desired.

In this regard, we propose gCubik+i as a new interactive platform that naturally interfaces a 3D display with a tabletop display. A virtual 3D aquarium application that showcases its interactivity potential is shown in Fig. 1.

Real-time interaction, provided by both the

3D display and the tabletop display, allows multiple users around the table to naturally pick up static 2D fish images, from the table into their own hands, and see them transforming into full parallax interactive 3D fish images. Once in their hands, users can share and interact with real-time animated 3D fishes supporting their discussions by pointing and handing over the display to other users.

The gCubik+i platform introduces two novel interaction paradigms to existing tabletop display applications: 1) natural switching between the shared working spaces of the table and the users’ hands, thus effectively combining the interaction modes of both them; and 2) transforming static 2D images into interactive 3D images that can be viewed and manipulated as if holding a real object.

In this paper, we first describe the components of the gCubik+i platform prototype, namely the gCubik 3D display and its technical

contributions: a) wide field-of-view Integral Photography; b) real-time rendering for integral photography; and c) real-time animation support architecture for application development. We then present the implementation of an interactive virtual 3D aquarium application, based on the gCubik+i platform, and discuss how it supports collaboration among a group of users. Finally, we gave an overview of the its conceptual design, its background and how it differs from previously proposed work.



Fig.1 Interactive virtual 3D aquarium using the gCubik+i platform

2 The gCubik+i platform

2.1 System configuration

The contribution of the gCubik+i platform is to enhance the gCubik 3D display[2], providing real-time interactivity by directly touching its faces, and interfacing it with a tabletop display. The platform also provides a real-time rendering and animation software module for easy application development.

Figure 2 shows a photograph and a schematic system overview of the gCubik+i prototype.

The system consists of the gCubik 3D display with touch sensors on all of its faces, a tabletop 32 LCD display with an infrared(IR) sensor, a control box for processing input and output of image data and sensors signals, a PC computer for image rendering and signal processing.

The gCubik 3D display is connected to the control box using a cable carrying VGA cables for the LCDs and USB cables for the touch sensors. The length of the cable is about 1m in order to allow enough flexibility for users to manipulate and interact with the display. The rendering PC outputs six VGA-resolution images and sends them for display into the gCubik via video distributors, in real-time. The tabletop display, together with the IR touch sensor, is used to detect the proximity of the 3D

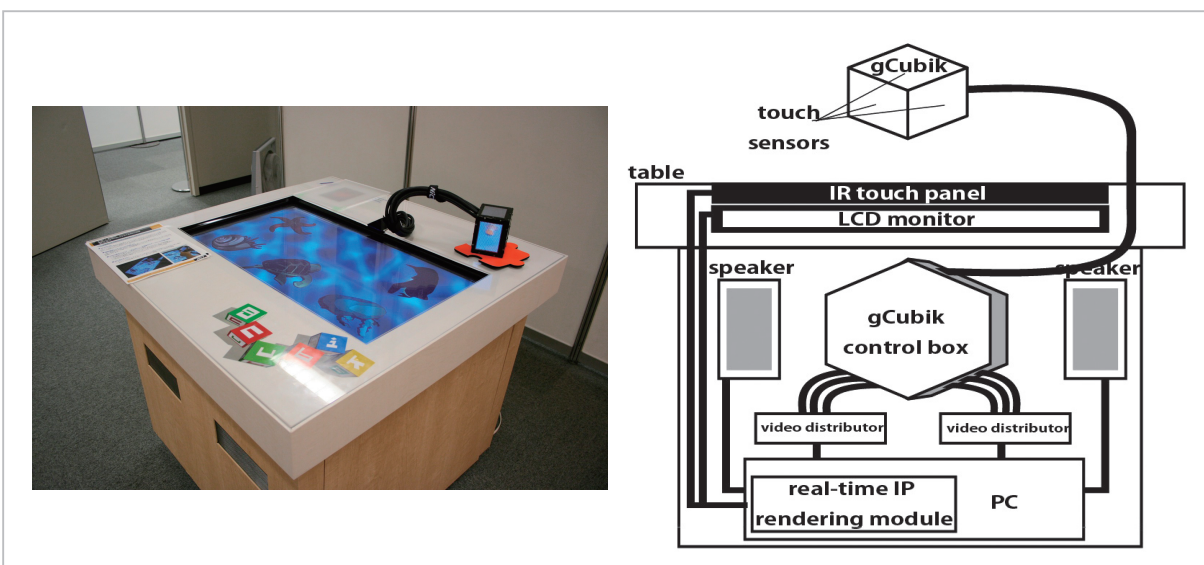


Fig.2 gCubik+i Platform Prototype

cubic display for interaction.

The following sections describe the hardware and software details of the gCubik+i platform.

2.2 gCubik: A glasses-free 3D display

We have proposed the gCubik[2], a graspable cubic 3D display, allowing multiple users to view and share 3D images without special glasses. The display provides stereo and full motion parallax images and can be viewed from any direction with correct viewpoint-dependent perspective. Users can grasp and manipulate 3D virtual objects inside the display with their own hands. The gCubik was designed to implement the “4g” part of the requisites for group-shared 3D display described later in section 4.1.

The display is a cube with 10cm per size,

weighing approximately 950g. Each face of the display is composed of an LCD panel, a micro-lens array plate, and a touch sensor, their layout is shown in Fig. 3. The micro-lens array provides the auto-stereoscopic viewing based on the Integral Photography (IP)[3][4] technique.

2.2.1 Wide field-of-view IP

Given that our cubic display is to be viewed from any direction, each IP lens should be able to modulate light rays at an angle of at least 2θ , as shown in the left part of Fig. 4. An extreme case occurs when simultaneously viewing three faces in display. Let D be the viewing distance, then the viewpoint becomes $e = \frac{D}{\sqrt{3}}(1,1,1)$. Also, let B be the size of one face and $c = (0,0,\frac{B}{2})$ be the center of the upper face F with normal $n = (0,0,1)$, then the above θ is the angle between the viewing vector $v = e - c$ and the normal n . For $D=400$ mm and $B=72$ mm, θ becomes 59.2 degrees.

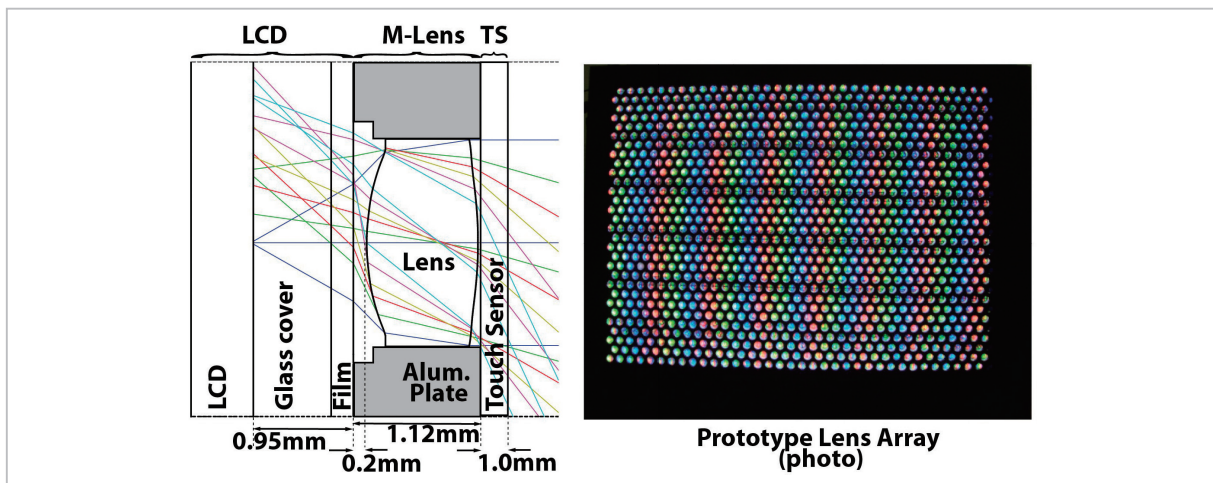


Fig.3 Prototype of our IP lens array

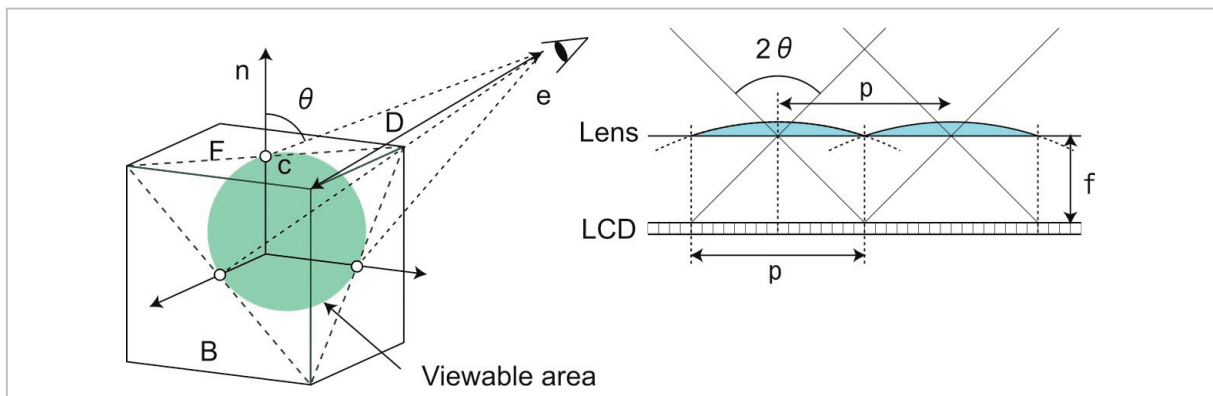


Fig.4 IP lens' field of view requirements

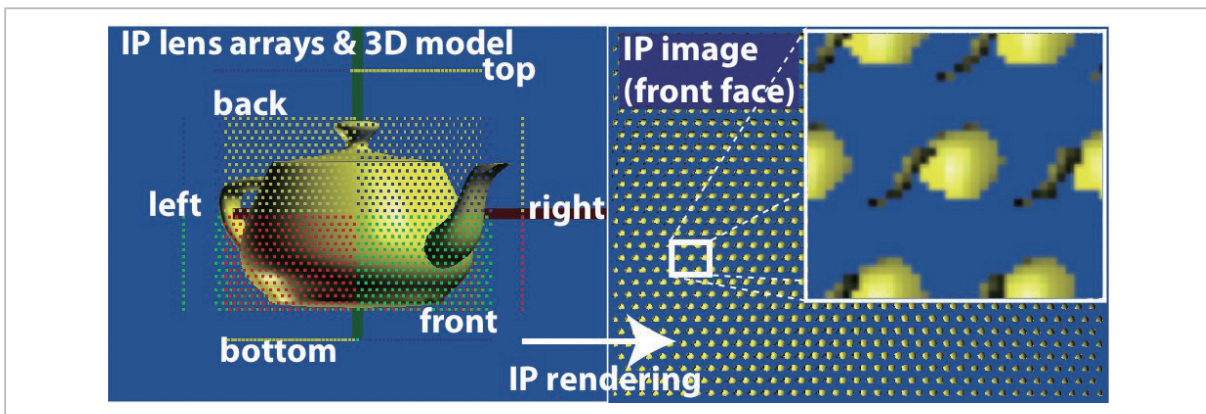


Fig.5 IP lens arrays and 3D model relative positions(left). Resulting IP image for the front face, zoom inset showing the details of the elemental images (right)

Therefore our cubic display requires an IP lens with a minimum of 120-degree field of view.

The type of lens array generally used in IP can be considered a planar array of several plano-convex lens of pitch size ' p ' tightly packed together as shown in the right part of Fig. 4. Commercially available IP lens have no more than 40-degree field-of-view, 2θ , thus not suitable for our purposes.

We designed and implemented an IP lens array with a 120-degree field-of-view. Each IP lens is a bi-convex lens arranged in front of the LCD on the left, as shown in Fig. 3. The IP lenses are mounted inside holes made on a 1.12-mm-thick black-painted aluminum plate. There are 1065 lens per plate. The holes have a radius of 2.0 mm and are arranged in a honey-comb layout to minimize the gaps between lenses.

The IP lens separation was fixed at 2.025 mm by setting each IP lens to cover a 18×18 -pixel area of the LCD display used in our experiments. The IP lens' focal length f , aperture and other parameters were optimized for image quality, and the LCD's glass and film were taken into account in the calculations.

2.2.2 Resistive touch sensors

Touch sensors were added to the gCubik on top of each of the faces to provide direct touch interactivity, refer to Fig. 3. Among the commercially available choices, we chose resistive-type touch sensors, versus capacitive-type ones, because they would trigger touch events only when users intentionally use their finger

nails for direct interaction. Capacitive-type ones would trigger touch events even when users were just holding the display with one hand.

Resistive touch sensors have two very thin metallic, electrically conductive layers. These layers affect the transmittance of the touch sensors, commonly around 80%, thus slightly reducing the brightness of the IP image. These layers may also slightly diffuse, and probably affect the stereopsis of the IP image. Quantitative measurement and necessary correction of the resulting IP image is part of our future work.

2.3 Real-time rendering for IP

The rendering algorithm utilizes OpenGL[5] off-screen rendering using framebuffer objects (FBO). It is an enhanced version of the algorithm in[6] and ideas from[7]. FBOs avoid context switching and copying from main memory to video memory thus improving performance. Also, the screen's dimensions do not limit the framebuffer object's dimensions. In our implementation, for each of the six cube's faces, we allocate an FBO of $(640 \times 2) \times (480 \times 2)$ pixels' dimensions, upsampling to avoid aliasing artifacts. And then render the 3D scene, into this FBO, from each of the 1065 IP lens positions using a 120-degree field of view virtual camera onto a $(18 \times 2) \times (18 \times 2)$ pixel viewport corresponding to each elemental image. We use a stencil mask to avoid viewport overlapping with adjacent lens' position. We then simply draw a down-sampled 640×480 pixel textured

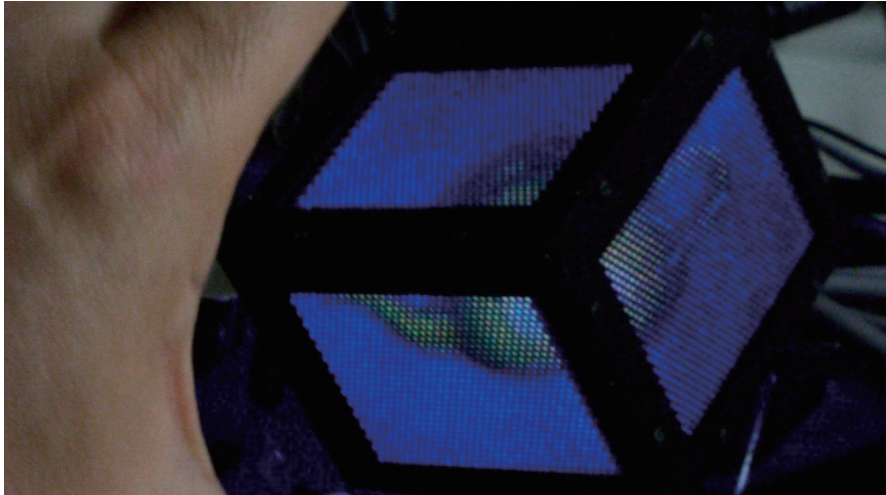


Fig.6 Simultaneous multi-face viewing for Integral Photography (teapot)

 Algorithm 1 : IP rendering algorithm using FBOs

```

Init Program
Initialize 3D Scene and sensors ## initGL()
Initialize FBO for each screen
...
Rendering Loop
update animation and sensors ## tickGL()
  for each screen in the cube display
    bind screen's FBO as rendering target
    for each elemental lens in the screen
      move camera to lens' position
      lookat inside the cube
      set camera projection to perspective
      set FBO's target rendering area ...
        ... according to lens position
      set circular/hexagonal mask
      render 3D scene to FBO ## drawGL()
    release screen's FBO as rendering target
    down-sample IP image to screen size
    display IP image on screen
  cleanup 3D scene and sensors ## closeGL()
End
  
```

quad on the window system provided framebuffer. The number of IP lenses in the array, 1065 per face, limits our display's 3D resolution to approximately 36×30 3D dots.

The left part of Fig. 5 shows the relative po-

sitions of the 3D model and the six IP lens arrays, corresponding to the six faces of the cube. The dot clouds represent the positions of each lens inside the lens array. The right part of the same figure shows the resulting IP image for

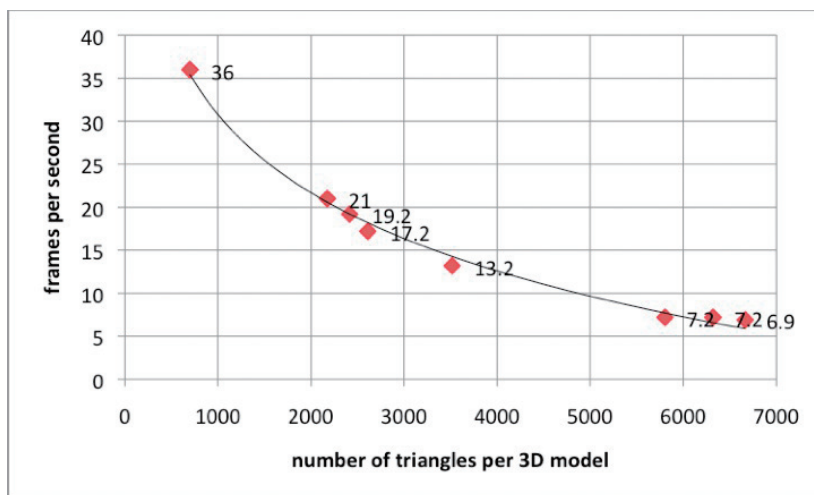


Fig.7 Proposed algorithm performance

the front face of the cubic display. The zoom inset shows the details of the elemental images generated. The elemental images are inverted due to the front face LCD display being positioned upside down in our prototype. Figure 6 shows a simultaneous multi-face viewing of a 3D teapot displayed on the gCubik utilizing the proposed wide field-of-view IP lens described here.

Algorithm 1 shows our proposed algorithm. During initialization, for each screen, we allocate an FBO with a texture and a depth buffer as images. This is the texture where the elemental images are generated. The rendering loop goes over all the screens of the display and over all the elemental lenses on each screen and moves a virtual camera to each lens position. Rendering the 3D scene using this camera then generates the IP elemental image.

Special care is needed to set the position of the rendering target area on the FBO by setting the viewport accordingly. Also, to avoid writing over other elemental images a scissor test is required. In case the elemental image shape is not a rectangle, a stencil mask should be used to avoid overlapping of elemental images. Once each screen's IP image is generated, the FBO contains such image, the algorithm simply draw a textured rectangle on the screen's framebuffer. Here, texture mapping down-samples the IP image using the texture minifying

function, antialiasing if necessary. The rendering loop then repeats the above steps until a halt.

2.3.1 Performance results

In our tests, we achieved up to 12 fps for a multi-material, non-textured, 3500-vertex, 6500-triangle, shaded 3D teapot model, using a 2.66GHz Intel Core2 Quad CPU with one NVIDIA GeForce 8800 GT graphics card.

Figure 7 shows the performance of the proposed algorithm for 3D models with different number of polygons.

2.3.2 Aliasing considerations

Computer-generated IP rendering algorithms often have to consider aliasing problems. The aliasing comes from the commonly low resolution of the elemental images. A solution would be to render the elemental images at a higher resolution and then downsample the resulting image using an antialiasing filter. However, given that every single pixel of an IP image is meaningful, care you be taken that resulting motion parallax is not lost after antialiasing.

In our experiments, we used a magnifying factor of 2 ($mf=2$) thus first rendering each elemental image at 36×36 pixels and then use a bicubic filter to downsample to the final 18×18 pixel resolution.

Figure 8, shows some of the antialiasing effects we obtained. Image (a) shows the elemen-

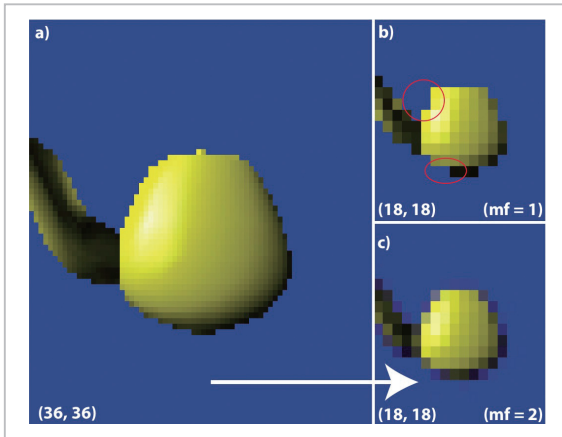


Fig.8 Elemental image aliasing issues.

(b) shows jagged-edges without magnification ($mf=1$). (a) is rendered using a magnification factor of 2 ($mf=2$) and used to produce the antialiased (c) by downsampling with a bicubic filter

tal image rendered at a higher 36×36 pixel resolution. Image (b) shows the elemental image by rendering as is at 18×18 pixel resolution. Note the jagged edges (aliasing) at the circled areas. These areas make the final appearance of the 3D object's shape different from the original one, when viewed in our gCubik display. Image (c) shows the resulting elemental images at 18×18 after downsampling image (a). Jagged edges are smoothed and the reconstructed 3D image show by our display looks closer to the original 3D object.

Antialiasing requires an extra use of FBO resources, limited by the available VRAM, as well as an extra computation step for downsampling. Even though downsampling/upsampling algorithms are quite optimized in most OpenGL's drivers, its abuse could have an impact on performance.

We tried different magnification factors, 4, 8 and 16, for rendering the elemental images to see if they affected the visual appearance of the final 3D IP image displayed on our gCubik. We didn't notice any major visual difference against our originally used $mf=2$. Indeed, we quantified the similarity of the images for each of the $mf=4, 8, 16$ against the one with $mf=2$ and found almost no difference. We used the root-mean-squared (RMS) of the difference be-

tween the two images and normalized it so that when a black image is compared to the reference image the similarity is 0, and when the images are identical, the similarity is 100%. Images with $mf=4, 8, 16$ are indeed very similar, around 98% to the one with $mf=2$. Therefore, we are assured that $mf=2$ is good enough for our purposes.

2.4 Real-time animation software architecture

The real-time rendering algorithm for IP described above effectively allows us to provide real-time animation and interactivity to our gCubik 3D display.

In order to ease application development for the gCubik 3D display, we implemented a plug-in architecture as shown in Fig. 9. We abstracted the IP rendering algorithm described above into a module just on top the OpenGL graphics library[5]. Application developers have the flexibility of adding real-time contents depending on their needs and expertise, via either a generic OpenGL plug-in, 3D model data or 3D model animation. They are all then passed down to the IP rendering module to generate the images required for the gCubik display.

The generic OpenGL plug-in allows experienced programmers to directly issue OpenGL commands to describe the application's virtual 3D world, its 3D objects and their behaviors. The developer's API, also shown in Algorithm 1, consists of four functions: *initGL* and *closeGL*, called only once during the application, mainly used to initialize or destroy the graphics context, load 3D model data, and any other sensor initialization necessary; *tickGL* called once during every loop of the IP rendering process, mainly used to update the virtual objects positions and behaviors according to sensor data inputs; *drawGL* is called every time an elemental image of the 3D scene is generated, as many as times as the number of IP lenses, 6×1065 in the case of the gCubik display. Therefore any performance optimizations should start by optimizing this function.

The 3D model data module can be used to

easily load and display 3D objects in different formats. The current implementation only supports the Alias Wavefront OBJ format but other format could be easily added. Graphics designers would use this module to easily display their 3D objects in the gCubik display.

The 3D animation module allows graphics designers and experienced 3D animators to develop animation in the 3D software of their choice. We natively support Quake MD2 animation format files. However, Maya animation files can be also be converted off-line into Quake MD2 animation files for rendering.

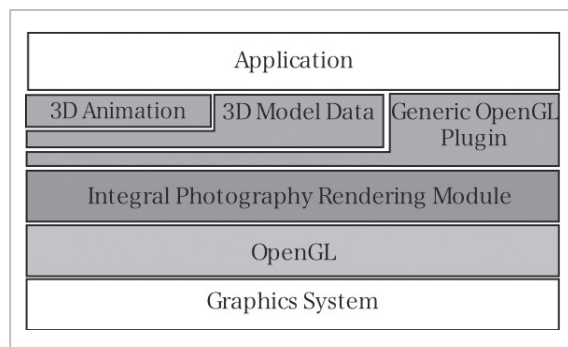


Fig.9 gCubik IP rendering software architecture, supporting real-time animation

3 Implementation and discussion

3.1 Virtual 3D aquarium application

In order to showcase the original capabilities of the gCubik+i platform we developed an interactive 3D Virtual Aquarium as a proof of concept application. Users can simultaneously view, share and hand over to others interactive 3D fishes, thus better supporting their group collaboration. This application enhances a similar one described in[8], by adding group-shared interactive 3D images.

Using the gCubik, a graspable cubic 3D display, users can naturally pick up static 2D fish images from a digital illustrated book on a table and see them transforming into 3D fishes. The 3D fishes in the display are interactive and animated in real-time. They are viewable from any direction without special glasses.

By dragging their fingers on the sides of the display, users can explore different viewing angles of the fishes. Users can also scare away the fishes by repetitively stroking on the faces of the cube. To release the fish back into the 2D digital book on the table simply touch its previous location with the gCubik. Figure 10 illustrates the flow of steps during the interaction.

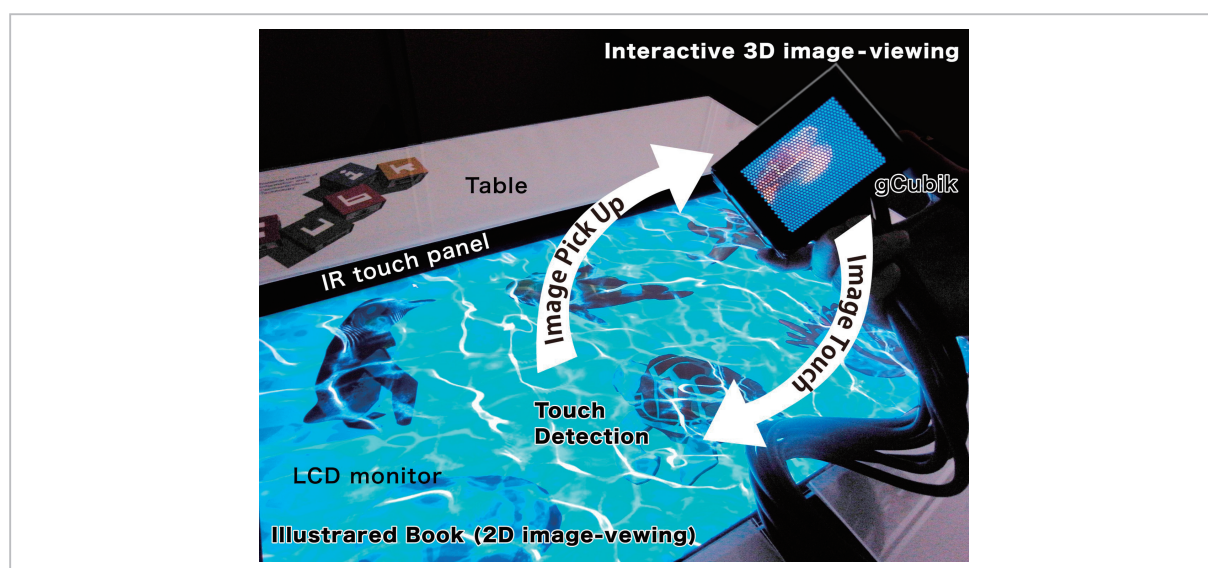


Fig.10 Interaction process and image exchange flow between the gCubik and a digital illustrated book

3.2 Users interaction experiences

The gCubik+i platform and its Virtual 3D Aquarium application have been exhibited in several major venues[9][10]. The following discussion is based on the analysis of video recorded sessions of users interacting with the system during the exhibits[9]. More than one thousand users interacted with the system during eight-hour four-day exhibits.

3.2.1 Group-sharing

Figure 11 shows a typical scene of users interacting with the system. We repeatedly observed the group sharing actions discussed in Section 4.2. Namely, 1) gather around the table, 2) simultaneous pointing, 3) handing over the display to others for closer inspection, and 4) natural eye contact while switching attention from the display to other users comments.

Gathering around the table (1) and simultaneous pointing (2) were possible given that our display can be viewed, in stereo, from any direction with correct perspective for any given viewpoint. Users would feel as if they were holding and looking at a real object.

Handing the display (3) to other users for closer inspection and natural eye contact (4) were possible because our display is compact and graspable. Instead of having users to stare at a shared flat screen as in traditional tabletop displays, users of the gCubik would bring the display closer for inspection and naturally hand it over to others if necessary.

The gCubik+i platform is perfectly suitable for collaborative tasks where turn-taking

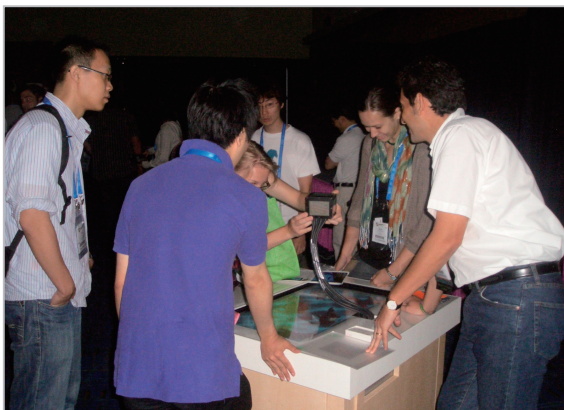


Fig. 11 Users interacting with the virtual 3D aquarium

among users would hinder the flow of interaction. The platform effectively integrates the four group collaboration requirements described above.

3.2.2 Natural 2D/3D display interface

Most users' comments concerned about how natural, direct and intuitive the interface between the 3D display and the tabletop display is. Occasionally users thought they were actually fishing by simply picking up the fishes from the table using the gCubik display, as if the fishes were actually jumping over when the gCubik approach them. Even users experienced in human-computer-interaction took some time to realize that the only sensor was the IR touch sensor on the table and not a special sensor in the gCubik. We believe this was due to the following factors of the interaction: the gCubik was the only moving pick-up device; the natural hand actions to pick up and release the fishes from the table to the gCubik; and by effectively removing the fish from the table and transferring into the gCubik display, instead of simply copying it distracts the attention of the users into the gCubik[11]. This reinforced our expectations of the gCubik+i platform concept, discussed in Section 4.3, of using the graspable 3D display as the natural device to pick up or release virtual objects on a table.

4 Design concept

4.1 Group-shared 3D display (4g+i)

The gCubik aims to become a natural replacement of real objects as part of a group discussion. Our prototype has the following design characteristics: *g1*) Group-shared: simultaneous viewing for multiple users with correct perspective regardless of their positions, *g2*) Graspable: easy to hold in one hand and pass it over to others, *g3*) Glasses-free: 3D stereo viewing without special glasses, *g4*) Glazed-showcase: give the impression of 3D real objects are inside a transparent case, and *i*) Interactive: provide real-time natural interaction via simple hand or finger gestures.

Figure 12 shows how such a 3D display would provide a shared object of discussion al-

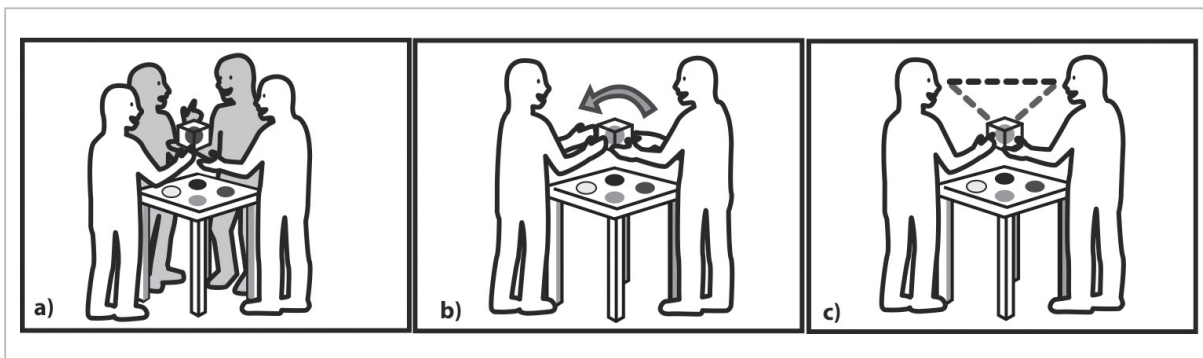


Fig.12 Feature requirements for group-shared 3D display

a) simultaneous viewing regardless of users positions; b) natural pointing and easy handing over of the object; and c) natural eye contact

lowing multiple users to simultaneously view the display regardless of their positions. Its compact size would allow easy handing over of the object as well as maintaining natural eye contact among the users during the discussion.

4.2 Background: Table-top collaboration

Much research have been done on the fields of computer supported collaborative work (CSCW) and human-computer-interaction (HCI), on how to facilitate and enhance the traditional way, a group of people would discuss and collaborate around the table. Here, we focus our discussion to those areas concerning, a) tangible users interfaces (TUI), b) multi-user tabletop interaction and c) Three-dimensional (3D) displays.

TUIs (a) promise to leverage our natural object manipulation skills by using physical artifacts as representations and controls for digital information. Physical objects become both the display and the interface to bits of data, thus making the interaction direct and intuitive, leading to “direct engagement[12].” For example, assembling small cubes into a structure as input interface for 3D model data into a computer[13][14], or touching real plants to input the different parameters to generate a virtual 3D plant[15].

Tabletop interactive displays (b) are commonly used to support group collaboration. Of special interest are the ones allowing multiple users to simultaneously interact with digital information by touching the display with their

bare hands[16]-[18]. A number of techniques for interfacing multiple displays that allow multiple users to share virtual objects by transferring them between 2D displays using a simple and intuitive pick-and-drop metaphor can be found in[19][20].

3D displays(c) promise to enrich the visual experience of interactive systems[21]. Examples varying from stereoscopic displays using special glasses[22], to auto-stereoscopic displays and volumetric displays[23]-[25], however, they are heavy or large and thus cannot be freely manipulated neither touched for direct interaction. Integral Photography (IP) is one of the techniques used for 3D displays[3]. A micro-lens array is used for recording and displaying multiple perspectives of a real scene. There is no need of special glasses; stereopsis and full parallax clues are readily available. Real-time extensions of IP to video have been proposed for medical applications[26] and live real scenes rendering[27][28]. Although these systems have enhanced viewing angles they are still designed for single-screen frontal viewing.

4.3 Natural interface between a tabletop and a 3D display

Consider the following scenario: a group of people is discussing about a new model of a cellular phone. If possible, somebody will almost immediately take his own cellular phone from his pocket, put it on the table, start pointing at parts of it while describing the features of the new model. He could also hold it and

hand it to any other person for a closer inspection. In this way, the discussion becomes more concrete and the group understanding of the topic becomes also smoother. This scenario is quite common on any face-to-face collaborative tasks among a group of people.

The gCubik+i platform aims to allow multiple users to naturally share and interact with 3D images, thus supporting group collaboration, by naturally interfacing a 3D display with a tabletop display. The 3D display would replace real objects with interactive virtual objects, whereas the tabletop display would allow selecting among different virtual objects. Figure 12 shows an interaction scenario illustrating this conceptual idea.

The interaction should be as natural as if users were picking up different objects from a table by simply grabbing them with their hands and bringing them to the center of the group for discussion. Ideally, the 3D display should serve as the displaying device as well as the device to pick-up/release virtual objects directly from the tabletop display. In this way, smooth switching between the shared working spaces of the table and the users' hands effectively combines their interaction modes enriching the collaboration.

Figure 12 shows how the 3D display itself should be used as the device to pick-up/release virtual objects on the table.

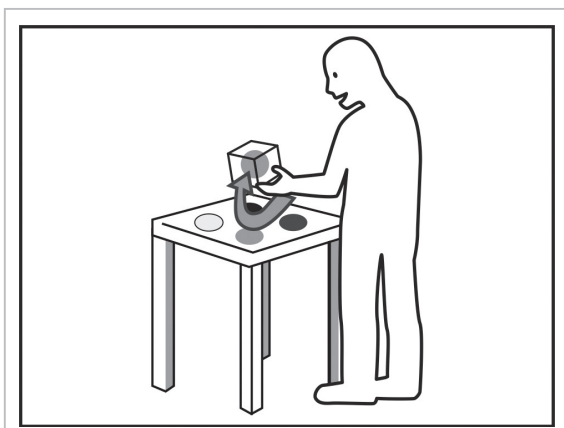


Fig.13 A graspable 3D display serving as the device to pick-up/release virtual objects from the table

4.4 Related work

In this section, we review and discuss previous work that closely relates to our proposed interaction platform. We focus mainly on graspable 3D displays and multiple-user tabletop systems.

Several cubic-shaped 3D displays have been proposed before, they are graspable and interactive. However some would require special glasses to produce a stereo effect[22], lacking the characteristic $g3$ above. Other proposed displays[29]-[32] do not require special glasses but need to track the user's head to provide 3D motion parallax, and are single-user by design. These displays can not simultaneously satisfy $g1$ and $g3$ above.

A number of interactive tabletop displays with multi-user capabilities have also been proposed. For example, the one found in[33] allows up to four users to simultaneously view and share stereo images with 3D motion parallax by tracking the users' heads. It consists of a stereo display along with a display mask with a hole in its center. However, users need to wear special glasses. Also, users can naturally point at the virtual objects but are not able to grasp them for direct manipulation. This system lacks $g2$ and $g3$ above. A somehow related system was proposed in[34], it uses a specially designed optical screen to provide multi-perspectives for different users. It also includes interactive small physical objects that serve as miniature screens displaying different objects. However, its current implementation only supports two users at fixed locations. This system implements all of the requirements in the previous sections, but only partially by restricting the number of users $g1$ and the manipulation of the objects happens only on top of the table $g2$. A glasses-free stereoscopic display using IP providing interactivity on a tabletop setup can be found in[35]. Multiple users can view the display in stereo and full motion parallax, however all the interaction happens on the top of the table over a single IP screen.

Our proposed gCubik+i platform promises to effectively integrate all the “4g+i” requirements discussed above.

5 Conclusions

In this paper, we have proposed gCubik+i as a new interactive platform that naturally interfaces a 3D display with a tabletop display, suitable for group collaboration. We outlined the conceptual design of the gCubik+i platform as well as implementing an interactive virtual 3D aquarium application based on this new platform. Preliminary observations of users interacting with the system suggest that the interaction is natural and intuitive, and supports group collaboration by allowing multiple users to view and manipulate virtual 3D objects as real objects.

We plan to use the gCubik+i platform to de-

velop multi-user applications where sharing 3D information would support collaborative tasks. Applications include, games, edutainment and net shopping advertisement among many others. We are primarily interested on research related to new interaction paradigms that may evolve from using the gCubik+i platform, and also performing evaluation experiments of its effectiveness compared to other types of interaction. We plan to integrate wireless image transfer between the gCubik and the rendering PC, thus easing free manipulation. We are also interested on research on a multi-gCubik interaction paradigm.

References

- 1 K. Enami, "Research on ultra-realistic communications," in Proceedings of SPIE, Vol. 7329, p. 732902, 2009.
- 2 S. Yoshida, R. Lopez-Gulliver, S. Yano, H. Ando, and N. Inoue, "gcubik: Implementation of a graspable cubic auto-stereoscopic display for multi-user environments," The Institute of Image Information and Television Engineers, Vol. 64, No. 4, pp. 118–124, 2010.
- 3 G. Lippmann, "Epreuves reversibles donnant la sensation du relief," Journal de Physique Theorique et Appliquee, Vol. 7, No. 1, pp. 821–825, 1908.
- 4 J. Arai, H. Kawai, and F. Okano, "Microlens arrays for integral imaging system," Applied optics, Vol. 45, No. 36, pp. 9066–9078, 2006.
- 5 D. Shreiner, M. Woo, J. Neider, and T. Davis, OpenGL(R) Programming Guide : The Official Guide to Learning OpenGL(R), Version 2 (5th Edition). Addison-Wesley Professional, August 2005.
- 6 M. Halle, "Multiple viewpoint rendering," in SIGGRAPH '98: Proceedings of the 25th annual conference on Computer graphics and interactive techniques, pp. 243–254, ACM, 1998.
- 7 F. de Sorbier, V. Nozick, and V. Biri, "Gpu rendering for autostereoscopic displays," in 3D Data Processing, Visualization and Transmission, 2008.
- 8 C. Sommerer and L. Mignonneau, "'A-Volve' an evolutionary artificial life environment," in Artificial life V: proceedings of the Fifth International Workshop on the Synthesis and Simulation of Living Systems, p. 167, The MIT Press, 1997.
- 9 R. Lopez-Gulliver, S. Yoshida, S. Yano, and N. Inoue, "gcubik: real-time integral image rendering for a cubic 3d display," in SIGGRAPH '09: ACM SIGGRAPH 2009 Emerging Technologies, ACM, 2009.
- 10 CEATEC, "<http://www.ceatec.com>," 2009.
- 11 N. Suzuki, T. Kamiya, S. Yoshida, and S. Yano, "A Basic Study of Sensory Characteristics toward Interaction with a Box-Shaped Interface," in Proceedings of the 13th International Conference on Human-Computer Interaction. Part II: Novel Interaction Methods and Techniques, p. 522, Springer, 2009.
- 12 H. Ishii and B. Ullmer, "Tangible bits: towards seamless interfaces between people, bits and atoms," in CHI '97: Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 234–241, ACM, 1997.

- 13 Y. Kitamura, Y. Itoh, and F. Kishino, "Real-time 3d interaction with activecube," in CHI '01: CHI '01 extended abstracts on Human factors in computing systems, pp. 355–356, ACM, 2001.
- 14 J. G. Sheridan, B. W. Short, K. V. Laerhoven, N. Villar, and G. Kortuem, "Exploring cube affordance: Towards a classification of non-verbal dynamics of physical interfaces for wearable computing," in In Proceedings of the IEE Eurowearable 2003; ISBN 0-85296-282-7; IEE, pp. 113–118, Press, 2003.
- 15 C. Sommerer and L. Mignonneau, "Interactive plant growing," *Ars Electronica*, pp. 408–414, 1993.
- 16 J. Rekimoto, "Smartskin: an infrastructure for freehand manipulation on interactive surfaces," in CHI '02: Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 113–120, ACM, 2002.
- 17 P. Dietz and D. Leigh, "Diamondtouch: a multi-user touch technology," in UIST '01: Proceedings of the 14th annual ACM symposium on User interface software and technology, pp. 219–226, ACM, 2001.
- 18 H. Koike and M. Kobayashi, "Enhanceddesk: Integrating paper documents and digital documents," in APCHI '98: Proceedings of the Third Asian Pacific Computer and Human Interaction, (Washington, DC, USA), p. 57, IEEE Computer Society, 1998.
- 19 J. Rekimoto, "Multiple-computer user interfaces: "beyond the desktop" direct manipulation environments," in CHI '00: CHI '00 extended abstracts on Human factors in computing systems, pp. 6–7, ACM, 2000.
- 20 M. A. Nacenta, D. Aliakseyeu, S. Subramanian, and C. Gutwin, "A comparison of techniques for multi-display reaching," in CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 371–380, 2005.
- 21 T. Okoshi, Three-Dimensional Imaging Techniques. 1976. H. Hua, L. D. Brown, and C. Gao, "System and interface framework for scape as a collaborative infrastructure," *Presence: Teleoper. Virtual Environ.*, Vol. 13, No. 2, pp. 234–250, 2004.
- 23 H. Kimura, T. Uchiyama, and H. Yoshikawa, "Laser produced 3d display in the air," in SIGGRAPH '06: ACM SIGGRAPH 2006 Emerging technologies, p. 20, ACM, 2006.
- 24 A. Jones, I. McDowall, H. Yamada, M. Bolas, and P. Debevec, "An interactive 360 light field display," in SIGGRAPH '07: ACM SIGGRAPH 2007 emerging technologies, p. 13, ACM, 2007.
- 25 J. Clar, "<http://www.jamesclar.com/>," 2002.
- 26 H. Liao, S. Nakajima, M. Iwahara, E. Kobayashi, I. Sakuma, N. Yahagi, and T. Dohi, "Intra-operative real-time 3-d information display system based on integral videography," *Lecture Notes in Computer Science*, LNCS 2208, pp. 392–400, 2001.
- 27 Y. Taguchi, T. Koike, K. Takahashi, and T. Naemura, "Transcaip: A live 3d tv system using a camera array and an integral photography display with interactive control of viewing parameters," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 15, No. 5, pp. 841–852, 2009.
- 28 T. Koike, K. Utsugi, M. Oikawa, M. Kobayashi, and M. Yamasaki, "Interactive autostereoscopic display with 60 ray directions," in SIGGRAPH '06: ACM SIGGRAPH 2006 Research posters, p. 147, ACM, 2006.
- 29 N. Kawakami, M. Inami, Y. Yanagida, T. Maeda, and S. Tachi, "Study and implementation of the object oriented display," *IPSJ SIG Notes*, Vol. 98, No. 9, pp. 79–84, 1998.
- 30 B. Lam, I. Stavness, R. Barr, and S. Fels, "Interacting with a personal cubic 3d display," in MM '09: Proceedings of the seventeen ACM international conference on Multimedia, pp. 959–960, ACM, 2009.
- 31 I. Stavness, F. Vogt, and S. Fels, "Cubee: a cubic 3d display for physics-based interaction," in SIGGRAPH '06: ACM SIGGRAPH 2006 Sketches, p. 165, ACM, 2006.
- 32 J. Djajadiningrat, G. Smets, and C. Overbeeke, "Cubby: a multiscreen movement parallax display for direct manual manipulation," *Displays*, Vol. 17, pp. 191–197, 1997.

-
- 33 Y. Kitamura, T. Konishi, S. Yamamoto, and F. Kishino, "Interactive stereoscopic display for three or more users," in SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques, pp. 231–240, ACM, 2001.
- 34 Y. Kakehi, M. Iida, and T. Naemura, "Tablescape plus: upstanding tiny displays on tabletop display," in SIGGRAPH '06: ACM SIGGRAPH 2006 Sketches, p. 12, ACM, 2006.
- 35 M. Kobayashi, M. Oikawa, T. Koike, K. Utsugi, M. Yamasaki, and S. Kitagawa, "Character interaction system with autostereoscopic display and range sensor," in IEEE Symposium on 3D User Interfaces, 2007. 3DUI '07, 2007.

(Accepted Sept. 9, 2010)



Roberto Lopez-Gulliver, Ph.D.
Expert Researcher, Multimodal Communication Group, Universal Media Research Center
3D Image Media and Display Technology



YOSHIDA Shunsuke, Ph.D.
Expert Researcher, Multimodal Communication Group, Universal Media Research Center
3D Image Media and Display Technology



MAKINO Mao
Expert Researcher, Multimodal Communication Group, Universal Media Research Center
Interface Design for Haptic Device's Contents Creation



YANO Sumio, Dr. Eng.
Senior Research Engineer, Advanced Television Systems Research Division, NHK Science and Technology Research Laboratories
Broadcast System, Visual Information Processing



ANDO Hiroshi, Ph.D.
Group Leader, Multimodal Communication Group, Universal Media Research Center
Brain and Cognitive Sciences, Multisensory Cognition Mechanisms, Multisensory Interfaces