Interactive Viewing of Panoramic Images *

K. Der, A. Basu

Computing Science
University of Alberta
Edmonton, Alberta
T6G 2H1

J. Reyda

Panoramic Viewing Systems Inc.
24 Riverside Crescent
Edmonton, Alberta
T5N 3M5

Abstract

A software system that allows users to interactively view immersive images is described. The immersive images capture large fields of view, typically 360° around for panoramic and panospheric™ [1] images, with vertical fields that vary from 70° to complete spheres. The input images were captured using various imaging sensors including panoramic cameras, conic mirrors, and fish-eye lenses. These images are inherently distorted due to the non-planar projections involved. The software simulates reality by mapping the captured image onto a 3D model, such as cylindrical for panoramic images and spherical for fish-eye and panospheric™ images, and then allows the user to interactively view the scene with the aid of a virtual camera. The innovative component in our work involves an inexpensive image acquisition system — Panospheric imaging — which allows users to obtain an almost spherical field of view.

1 Introduction

The Panoramic Viewing System developed here is a computer program to enable the user to interactively view a scene in any direction with the use of a virtual camera. It allows the user to pan left or right, tilt up or down, and zoom in or out. This type of system falls into a category of Virtual Reality (VR) in which the goal is to recreate the real world as convincingly as possible — as opposed to the category of VR that tries to convince the users that they are in another reality of the computer's making. The basic function provided by the software is to allow the user to view warped images of a scene while correcting the distortions. The images are warped because all the information in a 3D scene is packed into a 2D image; essentially, a non planar projection is being mapped to a planar projection. Panoramic imaging technology can be applied in many areas where conventional static images may be inadequate, or where synthetic computer models would be difficult or time consuming to build. A major problem with conventional Virtual Reality systems is the labour intensive process of building models. Also, rendering of the model would be computationally intensive. The technology we have developed allows the use of a simple generic 3D model and maps the captured image of the scene onto that model, thus not only simplifying the model building process, but also decreasing the computational complexity of the rendering process. This method does not involve extracting 3D models from static images and then manipulating and rendering the model.

2 Previous Work

2.1 Mosaicing

Mosaicing is a process by which a sequence of smaller images is patched into a bigger image[8]. Panoramic images can be generated using this technique. It has also been used as a means of compressing video of a scene[5]. Omnimax images have been created from multiple perspective views using elliptical weighted filters [3].

2.2 Panoramic Lens

Panoramic images can be captured using a special lens designed to capture a panoramic field of view. Many such optical blocks or lenses exist, one of which is based on the optic block in Figure 1. The lens block designed by Powell [6] is capable of projecting a full 360° cylindrical field of view into an annular format. The light rays, represented by the dotted lines, enter the optic and undergo a total of two refractions and two reflections before exiting.

2.3 Conic Mirrors

Conic mirrors have been used in a number of sensing applications where the desired field of view was

*This work was supported in part by Panoramic Viewing Systems Inc. (PVSI). Mr. Jerry Reyda (President, PVSI) provided several important input on designing the system.

[1] Panospheric is a registered trademark of PVSI.

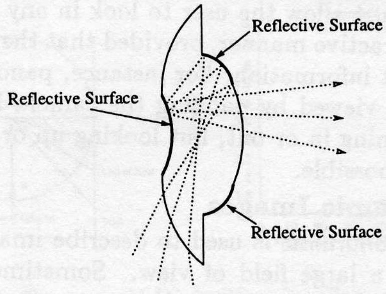


Figure 1: Panoramic Lens Block

larger than could be accommodated by a conventional camera. They have been used in mobile robots for obstacle detection and distance estimation [9, 10], and in pipeline inspection applications[7]. The basic principle is to use a reflective surface that is generated by revolving the desired profile about an axis. The profile of this reflective surface can be changed to give different performance characteristics, such as increasing or decreasing the field of view, or distributing the resolution to different portions of the image. Figure 2 illustrates how a typical conic mirror could be used to capture a 360^0 field of view.

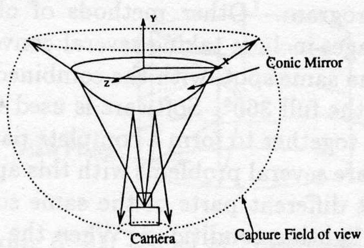


Figure 2: Conic Mirror Field of View

2.4 Panoramic Imaging Software

There are several panoramic imaging systems available commercially.

- **Microsoft Surround Video**
  The basic functionality provided is a cylindrical transformation of panoramic images, and it is implemented as an ActiveX control. Microsoft's surround video requires the use of an expensive special panoramic panning camera. The main disadvantage of this approach is the cost of the panoramic camera and the limited field of view which, like Apple's QuickTime VR, is only a cylindrical and not a spherical model.

- **Apple QuickTime VR**
  Apple's QuickTime VR system incorporates panoramic images into their QuickTime movies file format[4, 2]. The development kit includes software to stitch multiple conventional images to form a panoramic image. User intervention is necessary for some scenes in which there are few distinguishing features, the software will not be able to find the corresponding features to do the stitching.

- **Omniview**
  Omniview produces a product that will take two aligned fish-eye images and stitch them together to form a complete spherical view. The major disadvantage of this method is that the fish-eye lenses are very expensive and must be made to order.

2.5 Field of View

The field of view for panospheric^{TM} images is described as the percentage of the sphere surface which the captured image represents, as described by Bogner [1]. It was assumed that the field of view would be circular for all panospheric^{TM} images. We will generalize this and use it to describe the field of view for all camera models — including panoramic and panospheric^{TM}, as well as conventional and wide angle cameras. We will simply define the percent field of view as the ratio of the captured area on the surface of the sphere compared to the total surface area of the sphere. That is, we will relax the restriction of circular regions by allowing non circular regions as well.

2.5.1 Panospheric^{TM} Field Of View

Panospheric^{TM} field of view can be calculated by taking the ratio of surface area in the field of view on a sphere to the total surface area of the sphere. The percent field of view, as a function of the angle from the vertical axis, is plotted in Figure 3(a) and is calculated as

$$\%Field = 100 \times \frac{\theta(1 - cos\alpha)}{4\pi} \quad (1)$$

In this case, if we assume circular regions or $\theta = 2\pi$ then the $\%Field = 50(1 - cos\alpha)$. Figure 3(b) illustrates the region involved; as the angle α increases, the size of the *cap* increases and the field of view increases correspondingly.

2.5.2 Panoramic Field Of View

The field of view for panoramic images can also fit into this definition if we inscribe the cylinder in the sphere, such that the top and bottom of the cylinder are on the surface of the sphere and project the cylindrical

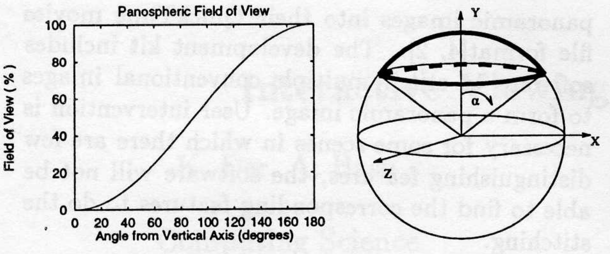


Figure 3: (a)Panospheric^{TM} Field of View Function (b)Panospheric^{TM} field of view corresponding to angle from vertical axis

surface onto the sphere. In Figure 4(b) the cylindrical surface is projected onto the sphere and is described as a function of the angle α which is a function of the ratio of the height and radius of the cylinder. The percent field of view (FOV) for panoramic (or cylindrical surfaces) can be calculated as

$$\%Field = 100 \times \frac{\theta sin(\alpha/2)}{2\pi} \quad (2)$$

where $0° < \alpha < 180°$ and $0° < \theta < 360°$.

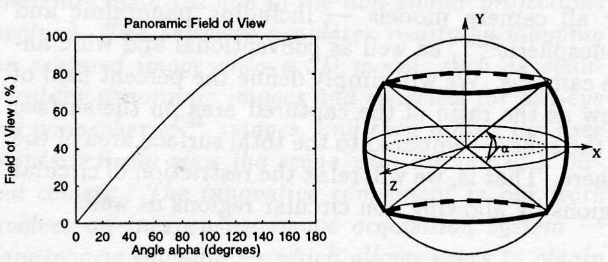


Figure 4: (a)Panoramic Field of View Function (b)Panoramic FOV corresponding to angle α

2.5.3 Distortions

When we capture panoramic or panospheric^{TM} projections onto images, we are essentially mapping these non geometric projections onto a geometric projection such as the perspective projection. The amount of perspective foreshortening is dependent on the FOV of the perspective or conventional camera. The wider the FOV the more apparent the distortions. Perspective foreshortening is more dramatic at the center of the viewing plane.

3 Viewing with a Virtual Camera

The desired system will have to take input images that are warped and do the appropriate transformations to correct the distortions and display the image as if it were viewed with a more conventional camera. The system must allow the user to look in any direction in an interactive manner, provided that the input image has that information. For instance, panoramic images can be viewed by panning the camera left or right and zooming in or out, but looking up or down would not be possible.

3.1 Panoramic Images

The term *panoramic* is used to describe images or pictures with a large field of view. Sometimes images with larger than 90° field of view are referred to as panoramic. We will use the term to mean truly panoramic or 360° field of view. Panoramic images are modeled as the surface of a cylinder. Consider a label on a can being a piece from a panoramic image. If we peel the label off and flatten it, we would have a remapped panoramic image which can be viewed on a planar display. These images will have a characteristic distortion due to perspective foreshortening.

Panoramic images can be acquired in many ways. There are panoramic scanning cameras that have a slit and, as the slit is revolved 360° about an optical axis, the film is exposed to the incoming light rays. Panoramic images can also be generated with computer graphics programs, such as ray tracing programs, that support panoramic cameras. Povray[2] is one such program. Other methods of obtaining panoramic images include taking several conventional images from the same spot, with the combined field of view covering the full 360°. Software is used to stitch several images together to form a complete panoramic image. There are several problems with this approach, one being that different parts of the same scene can have different lighting conditions. When the pictures are stitched together, the seams may be noticeable. It may also be difficult to match up the seams exactly. The problem of image mosaicing for panoramic applications was addressed by Szeliski [8]. Another way of generating panoramic images is from the fish-eye or panospheric^{TM} image formats.

3.1.1 Panoramic Texture Image Format

The texture format for panoramic images is a rectangular image, such as the one in Figure 7, which contains a full 360° field of view. Notice the panoramic distortions present in the image caused by perspective foreshortening.

Due to perspective foreshortening, objects that are further away will project to a smaller area than objects closer to the camera. In Figure 5(a) the camera is located at the center of the cylinder, as seen from

[2] povray can be obtained at http://www.povray.org

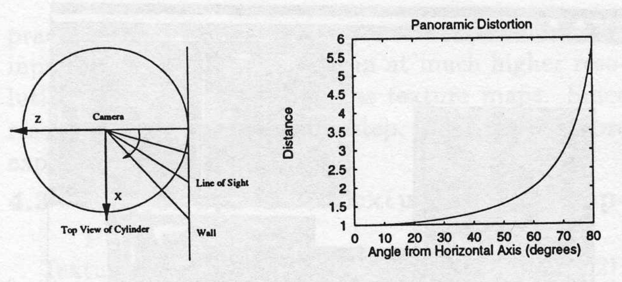


Figure 5: Panoramic Distortion (a) Top view of cylinder (b) Plot of distance to the wall

a top view. If we were looking at the wall in the diagram, we could plot the distance of the wall from the center of the cylinder as we look around in a clockwise direction (assuming that the cylinder has a unit radius). As we can see from Figure 5(b), the distance does not increase in a linear fashion. This implies that straight edges in the real world, that are not vertical when projected on the cylinder, would map to curved lines in the panoramic image. It is for this reason that viewing panoramic images with a smaller field of view, as in Figure 6, is not as simple as clipping the input image; we must correct the distortions. These distor-

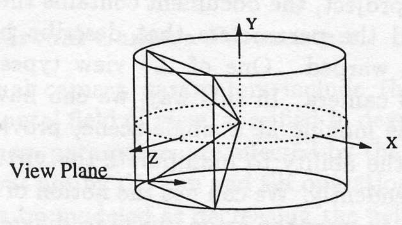


Figure 6: 3D Model of Panoramic Images

tions are more apparent in the panoramic images in Figure 7. The two images are shifted, to demonstrate

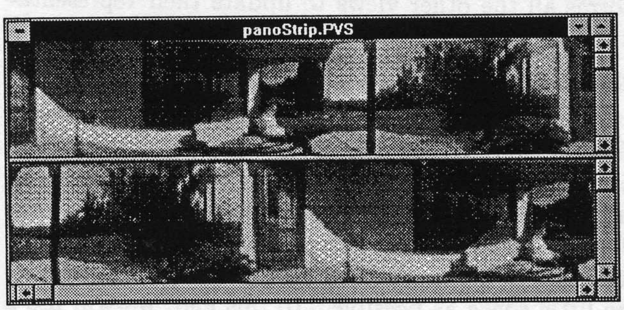


Figure 7: Panoramic Images

to the reader that each image is indeed a complete 360° around. Notice that the door in the lower image is wrapped around in the upper image. Note also that the shadow and the wall have the characteristic panoramic distortion due to perspective foreshortening.

3.2 Fish-eye Images

Fish-eye images are taken with a fish-eye lens, which captures a hemisphere of the scene; usually the resolution is poor at the edges and better towards the middle of the image. Figure 9 is an example of a fish-eye image[3]. Fish-eye lenses are expensive and are not mass produced.

The front and back fish-eye hemispheric image pairs are two calibrated fish-eye images taken with the camera pointing in opposite directions. Figure 8 illustrates the 3D model. The circular seam for the two halves is located in the XY plane.

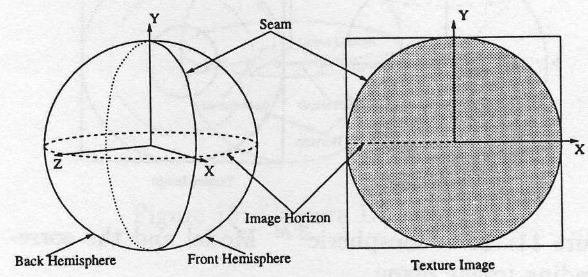


Figure 8: Front and Back Hemispheric Model

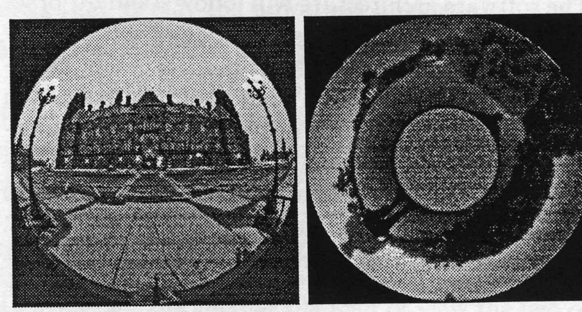


Figure 9: Left: Fish-eye image Right: Panospheric image

3.3 Panospheric^{TM} Images

The panospheric^{TM} optic being designed by PVSI is shown in Figure 10. This optic will provide a FOV larger than 70%.

The 3D model for panospheric^{TM} images is shown in Figure 11. Notice that the panoramic field from a to c is captured with the reflective conic section, and that the seam between the lens field and the reflect field is not continuous in the captured image.

[3] Image courtesy of Steve Bogner at Piercorp

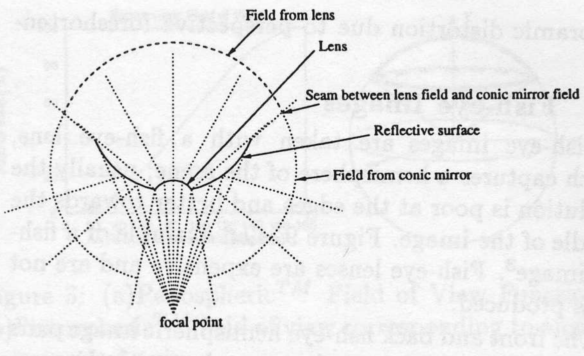


Figure 10: PVSI's Panospheric^{TM} Optic

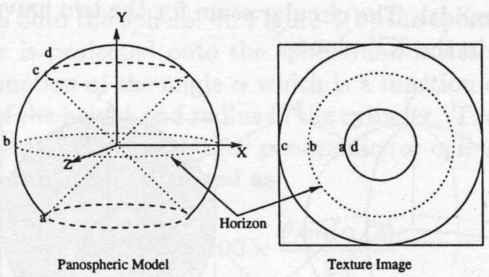


Figure 11: 3D Panospheric^{TM} Model and the corresponding image plane

4 Software Architecture

The software architecture will follow standard practices in use today by Microsoft windows software developers. It implements a Multiple Document Interface (MDI), where the main window of the application can contain more than one document and each document can contain more than one view window. Figure 12 illustrates the general (simplified) relationship between the application, documents, and the views. Figure 13 is an actual screen shot of the final software product that roughly corresponds to Figure 12. The application object is involved with the management and creation of the documents, and the main MDI frame window. The MDI Frame window manages all the MDI child windows, such as all the views

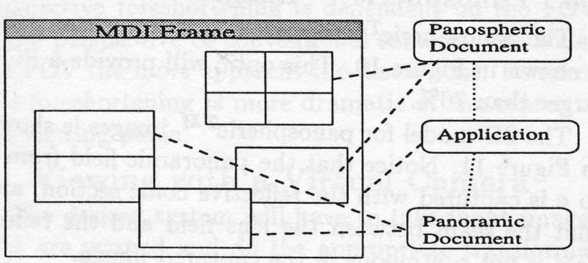


Figure 12: Multiple Document Interface

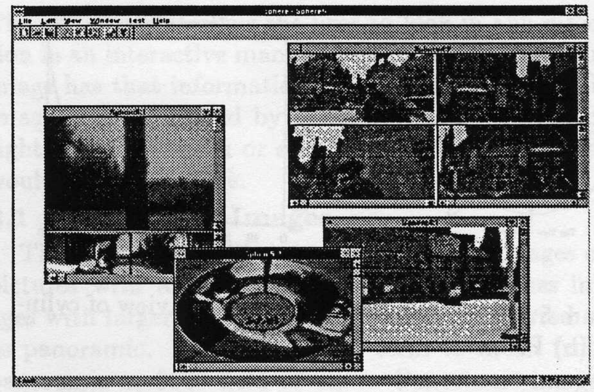


Figure 13: Multiple Document Interface Screen Shot

associated with the documents.

4.1 Document/View Model

The software is designed around the document/view model, which is used to encapsulate the data in the document and provide one or more views into the document. This is a generic object oriented approach that is commonly used to decouple the data from the way that it is presented to the user. For this particular project, the document contains the warped images and the parameters that describe how that image was warped. One of the view types will be our virtual camera. In this way, we can have multiple cameras looking at a single scene, providing the user with the ability to manipulate the virtual cameras independently. We can use the notion of different view types to view the same image in different ways, using different transforms. The views are the main method through which the user manipulates the document or the data. If the user changes the document data through a view object, the document object informs all the other views to update their representations accordingly.

4.2 File and Texture Image Format

There are two stages at which the texture image format should be considered: first, when it is stored in persistent storage such as a hard-drive or CD-ROM; second, when it is stored in main memory for transformation. When the images are stored on disk or transferred across a network we want them to take up as little space as possible. To this end, we will use a standard image format such as JPEG or GIF to compress and store the images. The texture images must be preprocessed or resampled to dimensions that are a power of 2. This restriction will facilitate much faster texture coordinate calculations, since bitwise shift operations can be used instead of multiplications. In practice this is not a severe restriction, because the input images can be scanned in at much higher resolutions than can be handled as texture maps. Since resampling is a preprocessing step, we can apply more expensive filtering techniques.

4.3 Object Space to Texture Space Mapping

Texture mapping is a method by which the flat 2D input images are mapped to 3D objects. The 3D objects are in turn transformed into a perspective correct view as seen by a virtual camera.

Generally, all the transforms are computed in the following manner:

1. determine the virtual camera parameters.
2. determine the appropriate 3D surface onto which to map the images (based on the selected 3D model).
3. determine the texture coordinates corresponding to the 3D surface points.
4. render the surface.

4.3.1 Virtual Camera Parameters

The virtual camera parameters include the vertical and horizontal field of view, specified in degrees or radians. These parameters are affected by the zoom operation, but not by the pan and tilt operations. Zooming in can be modeled as decreasing the field of view. Conversely, zooming out is modeled as increasing the field of view. In order to preserve the aspect ratio for the virtual camera, the vertical and horizontal fields of view are both multiplied by a scaling factor. Figure 14 shows the virtual camera located at the origin. We can see that by increasing the field of view, a larger surface area is visible to the camera. Because the camera's view is then mapped to a viewing window, which is of a constant size during the zoom operation, it is equivalent to zooming out. In other words, more of the scene will be visible, therefore the objects in the scene will be smaller. Each virtual camera view also contain

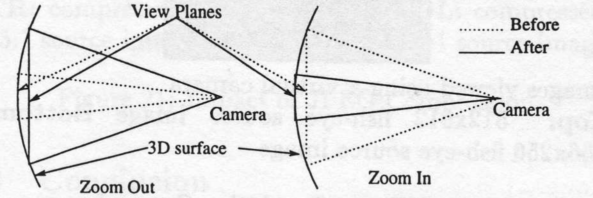


Figure 14: Zooming using a virtual camera

parameters for the ground plane angle and elevation angle. The camera direction vector can be specified in terms of two angles, as illustrated in Figure 15a. The ground plane angle is the rotation angle about the Y axis, and the elevation angle is the smallest angle between the camera direction vector and the XZ plane, or the ground plane. The camera direction vector actually represents the direction of the center of the camera. By changing the ground plane angle one would be effectively panning the camera. Similarly, by changing the elevation angle one could achieve tilting, or in effect make the camera look up or down.

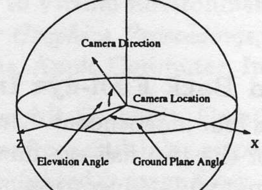


Figure 15: Camera Direction

5 Results

The interactive aspects of the software cannot be demonstrated on paper, but we can show some screenshots that display the unwarped images.

5.1 Panoramic Images

The image in Figure 16 was used to illustrate the distortion correction that the software performs. Notice that the straight vertical lines are still straight in the virtual camera views (Figure 17), but the horizontal lines are now curved due to perspective foreshortening. The amount of curvature required to correct for the distortion is dependent on the field of view. As expected, the smaller the field of view the less correction is required and the straighter the horizontal lines become.

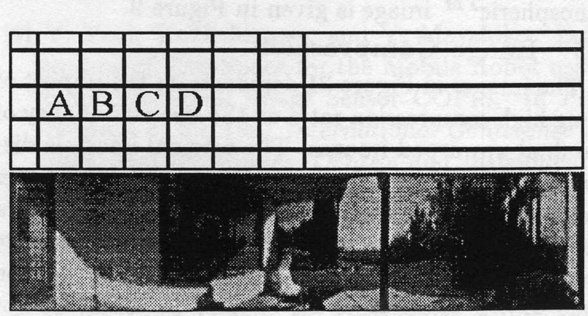
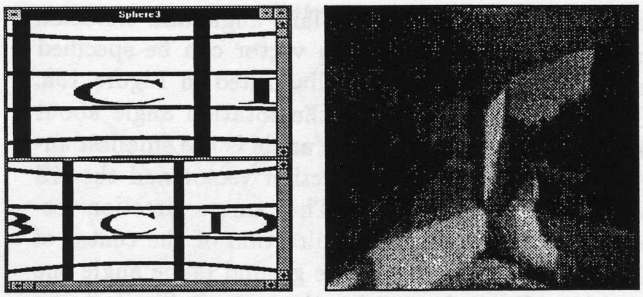


Figure 16: Input Panoramic Image



Top View: 40 degree Horizontal field of View
Bottom View:60 degree Horizontal field of View

Figure 17: Virtual Camera View of Test Panoramic Image

5.2 Front and Back Fish-eye Image Pairs

Figure 18 shows four independent views of the same scene composed of the two fish-eye images. Each of the views can be manipulated by panning, tilting, and zooming the virtual camera.

5.3 Input Image Resolution

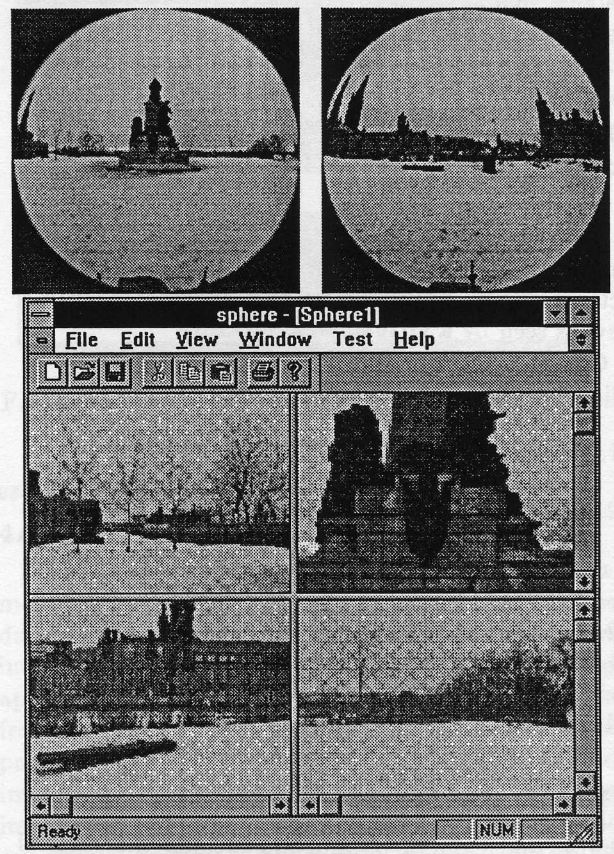
The input image resolution dictates, to a large extent, the quality that can be expected when scenes are viewed with the virtual camera. Since the views are usually of relatively small portions of the input image, pixelation can be very apparent. Figure 19 shows two views of the same scene, but the source images are of different resolution. The input image with higher resolution will produce better results. Although this may be obvious, the problem is more apparent because each texel is mapped to many pixels on the screen, thus reducing the natural filtering that occurs on monitors.

5.4 Panospheric^{TM} Image Mapped to a Panoramic Image

Figure 20 is a view of a panospheric^{TM} image transformed to a panoramic image. The virtual camera supports panning to the left or right. The source panospheric^{TM} image is given in Figure 9.

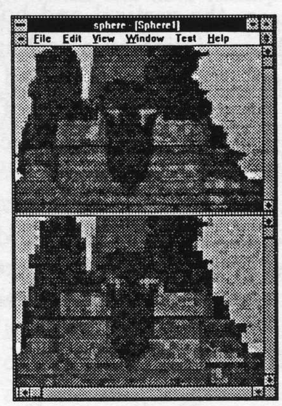
5.5 Image Compression

The images in Figure 21 demonstrate the impact of using high compression ratio on the size and quality of the final unwarped images. The original image is displayed in the upper left corner; the other three images are screen-shots of the unwarped conventional camera view. All the source images were 24 bit 1024x1024 images, each compressed to at least the compression ratio shown. Note that the checker patterns are not present in the screen display of the images; they were introduced in the conversion process to get it on paper. It may not be as apparent in the figure on paper,



Top Row: Fish-eye image pair
Bottom: Four independent views of the same scene

Figure 18: Views of Front and Back Fish-eye images



Images viewed using a virtual camera
Top: 512x512 fish-eye source image **Bottom:** 256x256 fish-eye source image

Figure 19: Different Resolution Source Images

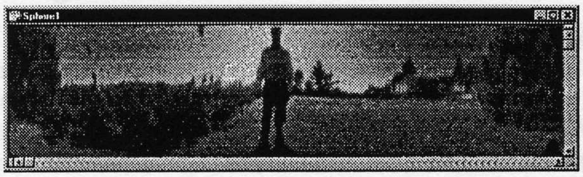
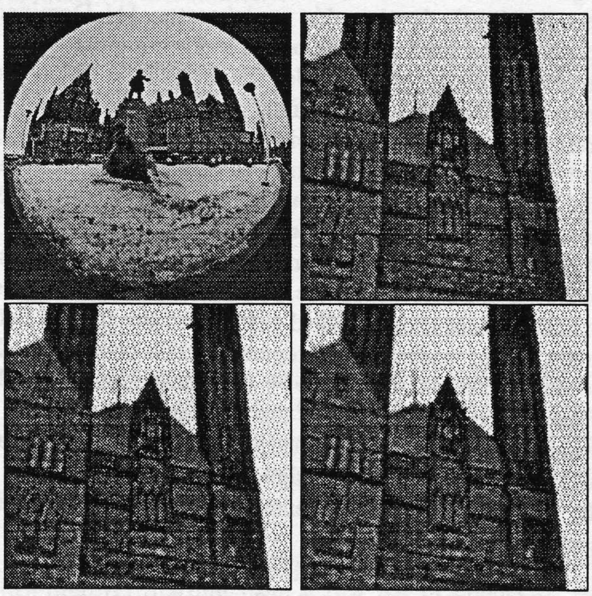


Figure 20: Panospheric^{TM} Image Mapped to a Panoramic Image

but on the screen we can notice very little difference between the results of the virtual camera viewing the 40:1 compressed image or the 55:1 compressed image. The quality degraded, however, when viewing the 80:1 compressed image, including loss of color information for the edges of the roof. This illustrates that we can compress the source images to a fairly small size and still get reasonable quality, even after the virtual camera viewing transform. For instance, the original uncompressed image was 1024x1024x24 bits or 3 megabytes; the largest of the three compressed source images was less than 80 kilobytes. However, these compression ratios are dependent on the content of the images.



TL: Original fish-eye image **Virtual view of:** **TR:** compressed 40:1 source image **BL:** compressed 55:1 source image **BR:** compressed 80:1 source image

Figure 21: Impact of JPEG Compression

6 Conclusion

Immersive imaging technology is a rapidly growing field with many potential applications. The ability to capture panospheric^{TM} images with the use of inexpensive conic mirrors, coupled with the ability to simulate interactive viewing with a virtual camera in software, will enable multimedia content creators to present virtual reality in a way not possible with computer generated graphics or static photographs alone.

References

[1] S. L. Bogner. An Introduction to Panospheric Imaging. *Proceedings of the 1995 IEEE International Conference on Systems, Man and Cybernetics*, 1995.

[2] S. E. Chen. QuickTime VR - An Image-Based Approach to Virtual Environment Navigation. In *Computer Graphics Proceedings, Annual Conference Series*. Apple Computer, Inc., 1995.

[3] N. Green and P. Heckbert. Creating raster Omnimax images from multiple perspective views using elliptical weighted average filter. *IEEE Computer Graphics and Applications*, pages 21–27, 6 1986.

[4] T.R. Halfhill. See You Around. *Byte*, pages 85–90, May 1995.

[5] M. Irani, S. Hsu, and P. Anandan. Video compression using mosaic representations. *Signal Processing: Image Communication*, v 7(n 4-6):p 529–552, November 1995.

[6] I. Powell. Panoramic Lens. *Applied Optics*, Vol. 33(No. 31):p 7356–7361, November 1994.

[7] D. Southwell, B. Vandegriend, and A. Basu. A Conical Mirror Pipeline Inspection System. In *Proceedings of the 1996 IEEE, International Conference on Robotics and Automation*, April 1996.

[8] R. Szeliski. Image Mosaicing for Tele-Reality Applications. In *Proceedings of the 2nd IEEE Workshop on Applications of Computer Vision*, pages 44–53, 1994.

[9] Y. Yagi, Y. Nishizawa, and Y. Masahiko. Estimation of Free Space for the Mobile Robot using Omnidirectional Image Sensor COPIS. In *Proceedings of the 1991 International Conference on Industrial Electronics, Control and Instrumentation - IECON'91*, 1991.

[10] K. Yamazawa, Y. Yagi, and M. Yachida. Obstacle detection with omnidirectional image sensor HyperOmni vision. In *Proceedings of the 1995 IEEE International Conference on Robotics and Automation*, 1995.