Immersive Panoramic Imagery

Mark Fiala
Computational Video Group, National Research Council, Ottawa, Canada K1A 0R6
mark.fiala@nrc-cnrc.gc.ca.

Abstract

An immersive experience where one views imagery captured from a panoramic camera with a head mounted display (HMD) or "cave" display system is an example of image processing that would be appreciated by the masses. In such a system, a user can see what would be seen from several viewpoints in a natural way by simply moving their head around. Virtual perspective views would be generated from recorded imagery collected by a panoramic camera from a set of locations. With image based rendering techniques, the user could also see views from viewpoints different from where the panoramic camera was placed. This paper proposes a simple framework for designing such systems based on *image cubes* which has the benefits of fast low latency operation and an efficient way to create intermediate images. The image cube method de-couples the image creation from the output image generation for the low latency required for realistic HMD immersive viewing. A fast algorithm for generating intermediate views along linear paths between capture sites based on pre-calculated disparity maps is also presented. A prototype system is shown.

Keywords: image-based rendering, immersive viewing, head-mounted-dispays (HMD's)

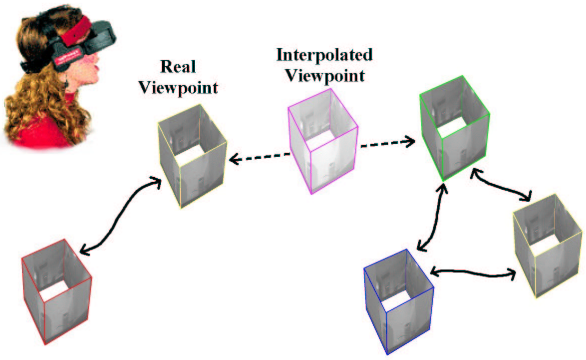


Figure 1: *Immersive viewing of panoramas. User can look around each panorama in the set, and move between them. The viewpoint is either at locations where the panoramas were captured or from panoramas interpolated with image based rendering.*

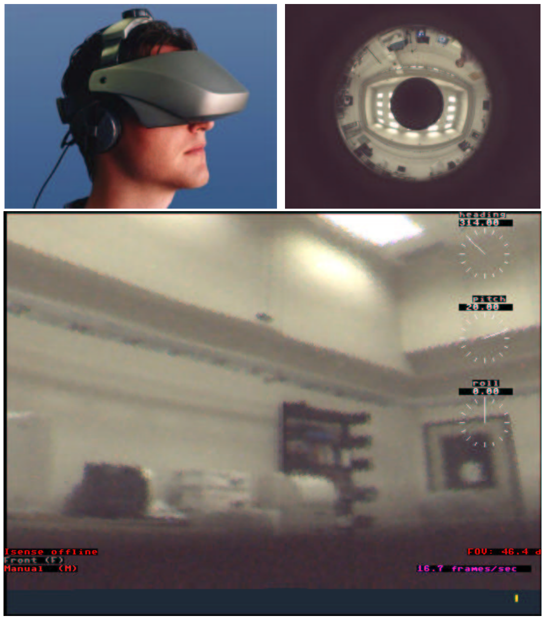


Figure 2: *Immersive viewing with an HMD (top left) of a section of a panorama generated from a panoramic camera image (top right). The user sees a pinhole projection model image (bottom).*

1 Introduction

This paper is motivated by the challenge of creating a system whereby a user can experience a scene using a head mounted display (HMD) or a cave (room with video or images projected on surrounding walls). A scenario where users can look around in a set of stored video or still frame panoramas is a powerful and entertaining way to experience a remote place, and would be useful to entertainment, "edutainment", virtual tourism, engineering analysis of sites, and many other applications.

A series of panoramic images (panoramas) would be captured at a series of neighbouring locations, in which the user could view a perspective view from one of these locations, or from positions between.

The user could look around as if they were present, and move to other real and interpolated spots with a joystick, mouse or keyboard press. The panoramas could also be created from computer generated environments rendered at high levels of realism, such as with ray tracing, that cannot be performed in real time.

To rapidly render the views and to create interpolated views without a detailed 3D model of the environment requires *image-based rendering* approaches. With an HMD, a perspective view would be warped from the stored or interpolated panoramic images according to the head position. This needs to have low latency to achieve a sense of realism, and so rendering must be fast. An efficient way of representing the panorama is important.

This paper details a system that achieves this with an *image cube* format for handling panoramas for fast rendering and rapid interpolation of intermediate views along straight line paths between panorama capture points.

2 Background

2.1 Panoramic Image Capture

Recent work has seen the use of non-perspective image projections for use in capturing images with a wider field of view. Omni-directional sensors can be built using a mixture of mirrors and lenses in the optical path, so called *catadioptric* cameras that can simultaneously capture light from a wider field of view than a conventional *dioptric* camera consisting of just a lens (or lenses) and a flat image plane. A panoramic camera, one that captures light from 360° along one axis, can be built using the combination of a standard dioptric camera and a rounded mirror. Basu [1] and others [2] have demonstrated such systems.

As shown in Fig. **??**, a vertically posed dioptric camera focused on a radially symmetrical mirror can capture light in all azimuth directions, with a viewing range above and below the horizontal horizon plane. Mobile robotics is one field that can benefit from such imagery presenting a continuous, simultaneous view from all azimuth directions. These images are formed radically different from traditional images that are usually only radially distorted perspective pinhole views, and they provide unique challenges for computer vision. One implication is the loss of line straightness in all directions except that parallel to the main camera/mirror axis (vertical in this discussion).

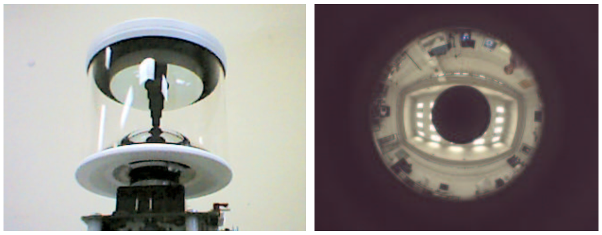


Figure 3: *A Panoramic Imaging system using Catadioptric optics and a sample panoramic image. The catadioptric camera is created by mounting a* NetVision Assembly B *mirror/lens unit onto a Vitana 1280x1024 IEEE-1394 digital video camera.*

2.2 Image and View Morphing

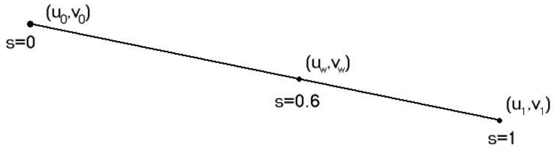
Seitz and Dyer [4] describe how to correctly find an intermediate view along the line between two captured views, the images should be rectified to a plane containing the focal points belonging to the two images. They show how *image warping* can be applied to obtain correct *view morphing* images.

Image warping is defined by Seitz *et al* as a linear mapping of a pixel location in an interpolated image between its position in the two original images. A pixel (u_w, v_w) in an intermediate image has coordinates calculated by linear interpolation between coords in the first (u_0, v_0) and second (u_1, v_1) image image.

$$w_o(P_0, s) = (1-s)p_0 + sC_{orr}p_0 = (1-s)p_0 + sp_1$$

$$u_w = (1-s)u_0 + su_1 \quad (1)$$

$$v_w = (1-s)v_0 + sv_1 \quad (2)$$



The following is a proof using image homographies that this defination of image warping will not work for the general case of two perspective views. A plane in space is imaged by both original images, these projections can be defined by homographies [3]. If the intermediate image does not have a projection of the plane that is also a homography, it is shown to not be a proper "shape preserving" intermediate image.

Consider points on a plane in the scene, world coordinates $(X, Y, 0)$. Image points can be expressed as homographies

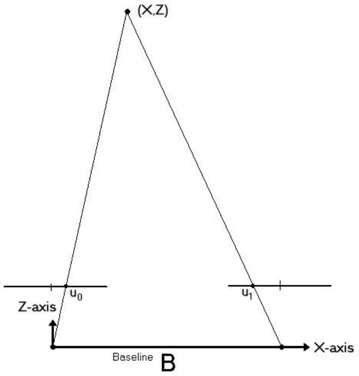
$$\begin{bmatrix} u_0 \\ v_0 \\ w_0 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$$
$$\begin{bmatrix} u_1 \\ v_1 \\ w_1 \end{bmatrix} = \begin{bmatrix} j_{11} & j_{12} & j_{13} \\ j_{21} & j_{22} & j_{23} \\ j_{31} & j_{32} & j_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \tag{3}$$

$$u_0 = \frac{h_{11}X + h_{12}Y + h_{13}}{h_{31}X + h_{32}Y + h_{33}}, v_0 = \frac{h_{21}X + h_{22}Y + h_{23}}{h_{31}X + h_{32}Y + h_{33}}$$
$$u_1 = \frac{j_{11}X + j_{12}Y + j_{13}}{j_{31}X + j_{32}Y + j_{33}}, v_1 = \frac{j_{21}X + j_{22}Y + j_{23}}{j_{31}X + j_{32}Y + j_{33}} \tag{4}$$

If we apply Eqn. 1:

$$\begin{aligned} u_w &= (1-s)u_0 + su_1 \\ = (1-s)&\frac{h_{11}X + h_{12}Y + h_{13}}{h_{31}X + h_{32}Y + h_{33}} + s\frac{j_{11}X + j_{12}Y + j_{13}}{j_{31}X + j_{32}Y + j_{33}} \\ &= \frac{k_1X^2 + k_2XY + k_3Y^2 + k_4X + k_5Y + K_6}{k_7X^2 + k_8XY + k_9Y^2 + k_{10}X + k_{11}Y + K_{12}} \end{aligned} \tag{5}$$

Eqn. 5 is not of the form $\frac{k_1X+k_2Y+K_3}{k_4X+k_4Y+K_6}$ and therefore image warping is not shape-preserving in the general case.



$$\begin{aligned} u &= f\frac{X}{Z} \\ u_0 = f\frac{X}{Z}, u_1 = f&\frac{X-B}{Z} = f\frac{X}{Z} - f\frac{B}{Z} \\ u &= (1-s)u_0 + su_1 \end{aligned} \tag{6}$$

The homography matrix is shown as a function of pattern origin (d_x, d_y, d_z) and plane axis $I_x = (i_xx, i_xy, i_xz), I_y = (i_yx, i_yy, i_yz), I_z = (i_zx, i_zy, i_zz)$.

$$\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_{center} \\ 0 & f_y & v_{center} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} i_{xx} & i_{yx} & d_x \\ i_{xy} & i_{yy} & d_y \\ i_{xz} & i_{yz} & d_z \end{bmatrix}$$
$$\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} = \begin{bmatrix} f_xi_{xx} + u_{center}i_{xz} & f_xi_{yx} + u_{center}i_{yz} \\ f_yi_{xy} + v_{center}i_{xz} & f_yi_{yy} + v_{center}i_{yz} \\ i_{xz} & i_{yz} \end{bmatrix}$$

If the displacement $\mathbf{D} = [d_x\ d_y\ d_z]^T = [B\ 0\ 0]^T$ then the image projection of the plane to the intermediate will be a homography. Only the horizontal U_w image position will change.

$$\begin{aligned} u_w = (1-s)u_0 + su_1 = (1-s)&\frac{h_{11}X + h_{12}Y + A}{h_{31}X + h_{32}Y + h_{33}} \\ +s&\frac{h_{11}X + h_{12}Y + B}{h_{31}X + h_{32}Y + h_{33}} \\ =&\frac{k_1X + k_2Y + K_3}{k_4X + k_4Y + K_6} \end{aligned} \tag{7}$$

3 Cube format

A panoramic image collects all or part of the light incident on a point in space. People typically think of such a data set as a spherical image, however this does not lend itself to efficient storage and handling. If a six-sided cube format is used instead virtual perspective images can be more readily handled. The cost of an increased storage space (nearly doubled, $\frac{6}{\pi} = 1.9$) over a spherical representation is offset by the benefits of fast rendering with standard graphics hardware and the ease of intermediate view generation (Section 4).

The view seen in the HMD screen is a perspective view that can see up to three cube sides at once, the view is rendered with simple texture mapping. Since the cube side images were created from a reprojection of a captured panoramic image or synthetic image, the user is unaware of the joints between cube sides. Fig. 5 shows an HMD view with and without an overlaid grid to visualize the cube sides.

4 Cube Interpolation

To create intermediate views, interpolated cube images are created. Since the panoramas are expressed as sets of perspective images, the sides of the intermediate cubes can be created with view morphing techniques from the sides of the two original cubes.

Section 4 shows how perspective images can be interpolated in a correct shape-preserving manner

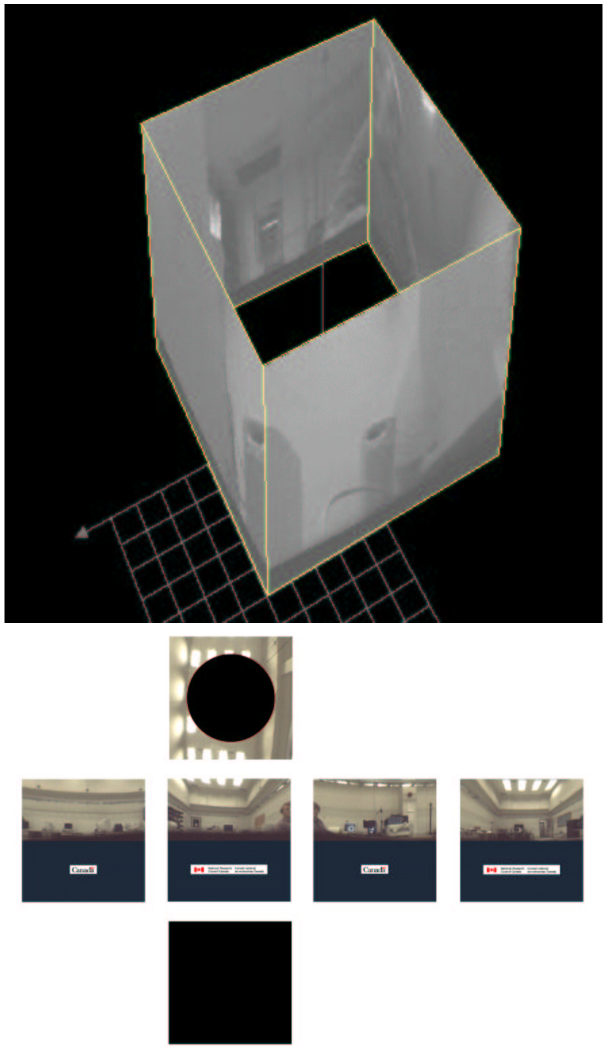
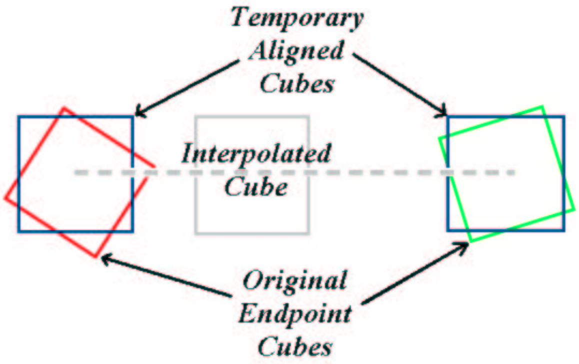


Figure 4: Cube image *format for panorama storage and handling. Panorama is stored as a standardized 6-sided cube. Top image shows visualization (4 sides only), bottom image shows 6 sides.*

Figure 5: *Immersive viewing with an HMD (top left) of a section of a panorama generated from a panoramic camera image (top right). The user sees a pinhole projection model image (bottom) in the HMD screen created from the* image cube *form of the panorama data.*

by first reprojecting the images to a parallel image plane. We perform this method on four of the six cube sides.

We propose reprojecting the two original cubes to intermediate versions of the same orientation as shown in Fig. 4 in a step we call *cube alignment*. If the orientation of the two cubes is known, then the reprojecting is trivial. In most cases it is not convenient to measure the poses of the cameras when the panoramas are captured. It is desirable to be able to just capture many images by simply moving the camera without measurements, and calculating from the images afterwards the relative orientation by computer vision analysis.



When the endpoint cubes are aligned so that an axis (X-axis herein) is parallel to the displacement vector in world coordinates, the displacement vector in the temporary cubes is $[100]^T$. The four sides parallel to the X-axis, "side" images, can be treated with image warping between their corresponding sides. Features in the intermediate cube side images will differ only along the X-axis. Objects in these "side" images will only translate in the X direction. Thus, the intermediate cube "side" images can be created by moving pixels along this axis according to a disparity map with only horizontal components. The two sides perpendicular to the X-axis, "front" and "back", will have their features in different places along radial lines from the center. The "front" image will have objects expanding out from the center while the "back" image will have objects contracting towards the center. These end images in the intermediate cube will be formed by moving pixels from or away from the centers according to a disparity map with only radial components.

The 4 "side" images of the interpolated cube can be found by a one-dimensional image warp of the corresponding sides of the aligned original endpoint cubes as per Section 4. Moving the image pixels according to the disparity map will create some holes, these need to be filled in some way to reduce their visibility to the user. The disparity map for each "side" image must be available, these are calculated offline along with the reprojection rotation matrices and provided with the set of cube panoramas. Disparity estimation procedures such as correlation take more processing and should be performed offline.

Depending on the quality of these intermediate cube images, they can be used for stationary viewing by the user, or just provided as quick transitions between cubes to visualize the direction travelled.

%vspace-0.15cm

5 Prototype System

Our first prototype system uses software rendering of up to 50 cubes. It was demonstrated on two HMD's; a I-Glasses SVGA Pro (29° diagonal FOV) and a Kaiser Electronics Proview XL50 (50° diagonal FOV) using imagery captured from the three panoramic cameras described in the next section, of which the Vitana/Remote-Reality Assembly B gave the best picture. The HMD orientation

The cube-viewing program can update the HMD screen at 16 Hz, it is being replaced with a version that uses the computer's video card graphics engine which will provide views at the full frame rate of the HMD display with reduced latency.

The cube viewing paradigm lends itself well to display in a cave environment, which was not yet deployed at the time of this writing.

6 Panoramic Camera Hardware and Panorama Quality

Three different omnidirectional cameras were tried in our system. They are all *catadioptric* sensors (contain both lenses and mirrors in the optical path) created by replacing the lens in a digital video camera with a commercial mirror/lens. The components and their useful image parameters are shown in Figure 6. The best image of the three systems was created using a Pixelink digital video colour camera (http://www.pixelink.com/)) fitted with a Remote Reality NetVision Assembly B panoramic lens/mirror assembly (http://www.remotereality.com/) (Figure 1.). It captures a color image of 1280x1024 pixels of which an annular region of 800 pixels diameter contains the panoramic image. The unused space is due to this model of lens/mirror assembly being designed for both $\frac{1}{2}$ and $\frac{1}{3}$ inch CCD's.

A system of similar pixel resolution was created using a colour 1024x768 pixel dragonfly IEEE 1394 camera, however the image noise was higher giving a poorer subjective perception. Finally an NTSC camera was used which gives the least high quality image, but allows for convenient use with an RF video link for our future intended tele-operated vehicle project.

With the Pixelink and Dragonfly IEEE cameras, providing a useful annular image of diameter 800 pixels, the perspective warp has an equivalent pixel density to a 320x240 image. With the narrow field of view (23.3° horizontal FOV) of our I-glasses HMD, the low resolution of the panorama is evident. It is the opinion of this author that catadioptric cameras are not available, at the time of this writing, which can provide a level of quality that a consumer system would demand. We are currently integrating a "Ladybug" multi-CCD IEEE 1394 camera from Point Grey Research which provides a nearly seamless hemispherical view with 6 separate cameras enclosed in a small package.

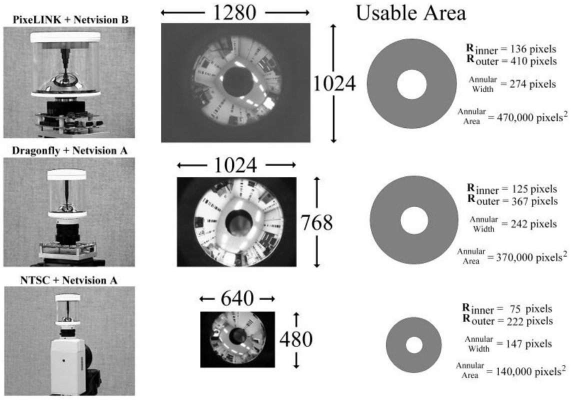


Figure 6: Resolution with three panoramic camera configurations.

7 Conclusions

An immersive viewing system was architected for building a system for viewing virtual views from recorded or computer generated rich panoramas, with an image-based rendering approach for creating intermediate views. This system would allow a user to navigate between captured views and see a view from that viewpoint using a head mounted display (HMD) or "cave" display system. This paper proposed a simple framework based on *image cubes* which has the benefits of fast low latency operation and an efficient way to create intermediate images. The image cube method de-couples the image creation from the output image generation to reduce the latency to improve HMD realism. A fast algorithm for generating intermediate views along linear paths between capture sites based on pre-calculated disparity maps was also presented. A prototype system under development was shown.

References

[1] A. Basu and D. Southwell. Omni-directional sensors for pipe inspection. In *IEEE SMC Conference*, pages 3107–3112, Vancouver, Canada, October 1995.

[2] M. Yamamoto H. Ishiguro and S. Tsuji. Omni-directional stereo for making global map. In *3rd Intl. Conf. Computer Vision*, pages 540–547, 1990.

[3] R. Hartley and A. Zisserman. Multiple view geometry in computer vision. Cambridge, UK, 2000. Cambridge University Press.

[4] Steven M. Seitz and Charles R. Dyer. View morphing. *Computer Graphics*, 30(Annual Conference Series):21–30, 1996.