Proc. of DARPA Image Understanding Workshop, New Orleans, May 1997

Generation of Perspective and Panoramic Video from Omnidirectional Video [*]

Venkata N. Peri and Shree K. Nayar
Department of Computer Science, Columbia University
New York, New York 10027
Email: {venkat, nayar}@cs.columbia.edu

Abstract

Existing software systems for visual exploration are limited in their capabilities in that they are only applicable to static omnidirectional images. We present a software system that has the capability to generate at video rate (30 Hz), a large number of perspective and panoramic video streams from a single omnidirectional video input, using no more than a PC. This permits a remote user to create multiple perspective and panoramic views of a dynamic scene, where the parameters of each view (viewing direction, field of view, and magnification) are controlled via an interactive device such as a mouse, joystick or a head-tracker.

1 Introduction

Remote visual exploration systems such as QuickTime® VR [Chen-1995] allow a user to navigate around a visual environment. This is done by simulating a virtual camera whose parameters are controlled by the user. A fundamental limitation of existing systems is that they are restricted to static environments, i.e. a single wide-angle image of a scene. The static image is typically obtained by stitching together several images of a static scene taken by rotating a camera about its center of projection. Only recently, it has become possible to acquire omnidirectional images at video rate (see [Nayar-1997]). The availability of such an acquisition device opens up the possibility of a software system that can create perspective and panoramic video streams. This adds a new dimension to the notion of remote visual exploration.

The omnidirectional camera developed by Nayar captures at video rate, a hemispherical field of view as seen from a single point. We have developed a real-time software system called OmniVideo that can generate multiple perspective and panoramic video streams from such an omnidirectional video stream. The user can create and orient multiple perspective and panoramic views in desired directions; all views are updated at video rate. Furthermore, the viewing direction, field of view, and magnification of each video stream can be controlled using an interactive device such as a mouse, joystick, or a head-tracker. The capabilities of the OmniVideo system can be exploited in a variety of applications, including immersive video, teleconferencing, autonomous navigation, and video surveillance and monitoring. We have also developed an omnidirectional web-camera wherein, view parameters can be modified using a control panel on the client's browser. An online demonstration is available at http://omnicam.cs.columbia.edu/.

2 The OmniVideo System

In the OmniVideo system, the omnidirectional video input defines the dynamic visual environment. Perspective and panoramic views are essentially virtual cameras positioned in this visual environment. Navigation and exploration of this visual environment is performed by modifying one or more camera parameters. Perspective and panoramic virtual cameras have five parameters, namely pan, tilt, zoom, roll, and field of view. In the OmniVideo system, the user can modify these parameters using an interactive device such as a mouse, joystick, or a head-tracker.

2.1 Reprojection

The optics of the omnidirectional camera is designed to reflect a wide-angle view orthographically off a parabolic mirror, onto the sensing element (CCD) of a conventional camera (see [Nayar-1997]). Views are generated by computing pixel intensities of every pixel $P(x_p, y_p, z_p)$ on the imaging surface of the virtual camera. Pixel in-

---

[*] This work was supported in parts by the DARPA/ONR MURI Grant N00014-95-1-0601, an NSF National Young Investigator Award, and a David and Lucile Packard Fellowship.

tensities are determined by reprojection. This is equivalent to determining the intensity of the point of intersection of the ray $R(\theta,\phi)$ (θ and ϕ are polar and azimuthal angles, respectively) from the focus of the parabolic mirror, in the direction of the point P.

The equation of the parabolic mirror is given by

$$z(r) = \frac{h^2 - r^2}{2h}, \quad r^2 = x^2 + y^2, \quad h > 0\,, \tag{1}$$

where h is the *parameter* of the parabola. The ray $R(\theta,\phi)$ intersects the parabola at a distance

$$\rho = h/(1+\cos\theta) \tag{2}$$

from the focus. When projected orthographically on to the CCD, the coordinates of the point of intersection are given by

$$x_i = \rho\sin\theta\cos\phi, \quad y_i = \rho\sin\theta\sin\phi\,. \tag{3}$$

Interpolation is used to determine intensity at this point.

Rewriting equation (3) we get

$$x_i = \frac{h}{z_p + \sqrt{x_p^2 + y_p^2 + z_p^2}} x_p\,, \tag{4}$$

$$y_i = \frac{h}{z_p + \sqrt{x_p^2 + y_p^2 + z_p^2}} y_p\,. \tag{5}$$

This form of equation (3) is suitable for optimization, as we shall see later.

2.2 Implementation

Video-rate performance is the most important feature of the OmniVideo system. Since the incoming visual information changes dynamically, OmniVideo *cannot* take advantage of most real-time reprojection methods that have been developed for static images (see [Chen-1995], [McMillan and Bishop-1995], [Lippman-1980], [Miller and Chen-1993]). We have implemented several numerical and data optimizations, which give to video-rate performance on a PC.

We use the notion of *geometric maps* to generate views. The geometric map defines the coordinate transformation between pixels on the imaging surface of the perspective (or panoramic) virtual camera, and the omnidirectional image. Simply stated, the geometric map implements the coordinate transformations of equations (4) and (5) in a lookup table. The process of reprojection is reduced to a lookup through the geometric map. The geometric map of a view changes only when viewing parameters of the associated virtual camera change. In computing views at video-rate, we observe that geometric maps provide the greatest speedup when viewing parameters are unmodified.

Yet another speedup is the use of lookup tables for geometric map generation. In equations (4) and (5), the term $\sqrt{x_p^2 + y_p^2 + z_p^2}$ represents the *distance* of the pixel from the focus of the parabola. As we shall see, this *distance* factor can be rewritten in a manner that is independent of all view parameters, except zoom, and hence can be determined from a lookup table.

For a perspective view, this *distance* can be written as $\sqrt{x_v^2 + y_v^2 + f^2}$, where (x_v, y_v) are the coordinates of the pixel $P(x_p, y_p, z_p)$ in the coordinate system of the virtual camera. f is the focal length of the perspective view. Substituting, equations (4) and (5) can be written as:

$$x_i = \frac{h}{z_p + \sqrt{x_v^2 + y_v^2 + f^2}} x_p\,, \tag{6}$$

$$y_i = \frac{h}{z_p + \sqrt{x_v^2 + y_v^2 + f^2}} y_p\,. \tag{7}$$

Similarly, for a panoramic view, the *distance* factor $\sqrt{x_p^2 + y_p^2 + z_p^2}$ can be expressed as $\sqrt{w^2 + f^2}$, where w is the *height* of the pixel along the cylindrical surface of projection, and f is the focal length (radius of the cylinder). Again, equations (4) and (5) become:

$$x_i = \frac{h}{z_p + \sqrt{w^2 + f^2}} x_p\,, \tag{8}$$

$$y_i = \frac{h}{z_p + \sqrt{w^2 + f^2}} y_p\,. \tag{9}$$

The *distance* factor now is effectively a constant. Using the optimized equations for x_i and y_i (equations (6), (7), (8) and (9)), it is possible to compute geometric maps of perspective and panoramic views at video rate.

Since reprojection of pixel coordinates (x_p, y_p, z_p) takes place in a raster scan manner, there is tremendous computational redundancy in inter-pixel computations. We exploit this redundancy by incremental computation of pixel coordinates.

A critical implementation issue, especially in video applications, is the overlap between computation, user interaction, and video display. OmniVideo takes advantage of the multithreading available in most modern operating systems (such as Windows® NT) to provide a responsive user interface, while operating at full video-rate.

Figure 1: The OmniVideo system allows a user to generate multiple perspective and panoramic video streams from an incoming omnidirectional stream (top-left).

3 Results

We have implemented OmniVideo on an IBM compatible Pentium Pro PC, operating at 200 MHz. The system has a simple interface that allows the user to control viewing parameters using either a joystick or a mouse. In this configuration, OmniVideo can generate up to 12 perspective and panoramic video streams at video rate. Figure 1 shows the system in a typical surveillance and monitoring application.

A novel application for the OmniVideo system is an omnidirectional web-camera. The OmniVideo system is integrated with an http server (such as Microsoft's Internet Information Server). A live omnidirectional camera feeds into the OmniVideo system. Multiple users can connect to the web-camera and navigate the scene captured by the omnidirectional camera, in real time over the Internet. View parameters are controlled using a control panel provided on the client's browser. The server running on a 200 MHz Pentium Pro PC can support a large number of connections at video-rates. An online demonstration is available at http://omnicam.cs.columbia.edu/.

References

[Chen, 1995] S. E. Chen. QuickTime VR – An Image Based Approach to Virtual Environment Navigation. *Computer Graphics: Proc. Of SIGGRAPH 95,* pages 29-38, August 1995.

[Lippman, 1980] A. Lippman, Movie Maps: An Application of the Optical videodisk to Computer Graphics, *Proc. Of SIGGRAPH 80,* 1980.

[McMillan and Bishop, 1995] L. McMillan and G. Bishop. Plenoptic Modeling: An Image-Based Rendering System. *Computer Graphics: Proc. Of SIGGRAPH 95,* pages 39-46, August 1995.

[Miller and Chen, 1993] G. Miller and S. E. Chen. Real-Time Display of Surroundings using Environment Maps. *Technical Report No. 44, Apple Computer, Inc.,* 1993.

[Nayar, 1997] S. K. Nayar. Omnidirectional Video Camera. *Proc. Of DARPA Image Understanding Workshop,* May 1997.