Synthetic Movies Derived from Multi-Dimensional Image Sensors

by

V. Michael Bove, Jr.

S.B. Electrical Engineering, Massachusetts Institute of Technology, 1983
S.M. Visual Studies, Massachusetts Institute of Technology, 1985

Submitted to the Media Arts and Sciences Section
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 1989

Signature of Author _____
Media Arts and Sciences Section
April 3, 1989

Certified by _____
Andrew Lippman
Lecturer, Media Technology; Associate Director, Media Laboratory
Thesis Supervisor

Accepted by _____
Stephen Benton
Chairman, Departmental Committee on Graduate Students

Synthetic Movies Derived from Multi-Dimensional Image Sensors

by

V. Michael Bove, Jr.

Submitted to the Media Arts and Sciences Section
on April 3, 1989, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Consider the case of a computer-animated movie. The individual frames of the movie are less compact and modifiable than the database and script that produced them. Ideally, when a movie is made of a real scene, the output of the camera should not be frames but rather a three-dimensional database describing the objects in the scene and their movements through time, as this would permit great data compression as well as computer-graphics-style operations and interactive modification of the movie.

It is demonstrated that lighting, focus, viewpoint, and other changes can be made to real scenes when they are captured by a camera capable of sensing range information. A process is developed permitting the estimation of range by use of focus gradients. This process is combined with a three-dimensional spatiotemporal gradient motion estimator employing a parametric signal model, so that the motions of objects may be tracked through time, and also to provide an additional, independent source of range data. A complete volumetric database is built up of visible portions of a moving scene over time so that it may be rendered from viewpoints other than that of the camera without missing parts.

The work described in this thesis has been supported by CPW Technologies and International Business Machines.

Thesis Supervisor: Andrew Lippman
Title: Lecturer, Media Technology; Associate Director, Media Laboratory

Contents

Chapter 1

Introduction

Now if you will venture to go along with me and look down into the bottom of this matter, it will be found that the cause of obscurity and confusion...is threefold. Dull organs, dear sir, in the first place. Secondly, slight and transient impressions made by objects when the said organs are not dull. And, thirdly, a memory like unto a sieve, not able to retain what it has received.

Laurence Sterne

The Life and Opinions of Tristram Shandy

1.1 The camera as model-maker

Movies – by which term is meant the whole spectrum of photographic and electronic systems by which images of moving subject matter are captured, stored, and displayed – are currently shot with cameras that take in a series of of two-dimensional projections of three-dimensional reality. This series of snapshots, or frames, is more-or-less preserved all the way from the camera to the viewer's eye and brain, where (fortuitously for

moviemakers) visual processing recreates the impression of continuous motion.

In the conceptually similar process of traditional animation, each frame must be drawn by hand. Computer animation, though, permits the manipulation of a three-dimensional database describing objects and surroundings, which are projected into two-dimensional frames (as by a camera) only at the final stage of the process. This method of operation, as Zeltzer has noted, can free the animator's energy from drawing key frames and in-betweening, and allow a greater emphasis to be placed upon creating environments and manipulating them [Zeltzer, 1985]. Computer graphics research has long worked with models of physical reality, and now the artist has at his or her disposal methods for modeling objects, rendering them from variable viewpoints with variable lighting, segmenting synthetic scenes and rearranging the constituent parts, adding and deleting objects, and so on. But a problem exists in obtaining interesting, realistic imagery on which to apply these methods. The time saved on in-betweening now goes toward converting the animator's visions into the mathematically-tractable object representations used by most computer-graphics software. For some totally imaginary objects this may be necessary, but for many real-world objects it would be useful for the computer somehow to "look" at them and incorporate them into animation databases.

The traditional moviemaker[1] is also concerned with creating environments and manipulating them. Moviemakers know well, probably better than computer graphics

[1]The practitioner of *cinema verite* might disagree with the terms "creating" and "manipulating," but the remainder of the comments below still apply.

artists, the effects careful lighting, depth-of-field and focal length selection, framing, and scene composition can have, and may go to great lengths during a shoot to adjust all these parameters to best effect. Ideally, the moviemaker would like to have the freedom of the computer animator, and be able to control such image variations in post-production. But movies, whether from a film camera or a video camera, are not in a "language" that computers can understand. Thus what is sometimes called "electronic cinematography" generally involves operations at the frame level: superimposing parts of frames, warping their shapes, and other such throwaway effects common to television news broadcasts.

A current research area of the Movies of the Future project at the MIT Media Laboratory is the use of adaptive vector coding to reduce the bandwidth of a moving, color image sequence to the point that it may be played back at standard compact disc data rates [Lippman, 1987]. The importance of this work is not merely in the bandwidth reduction, however, but more significantly in that the "CD movies" are not treated strictly as sequences of frames. Rather, treating a movie as a digital data stream opens up the possibility of rapid, useful interaction with sequences via computer and (given appropriate organization of the information) even within frames.

Just as the databases used to generate animated computer graphics are far more compact than the frames of the resulting animation, it ought to be possible to proceed in the opposite direction, reducing the frames of a real movie to a set of objects defined once and given three-dimensional trajectories. Such databases are also typi-

cally not tightly coupled to the spatiotemporal resolution (that is, number of pixels, and frames per second) of the output device upon which the scene will be rendered, rather, they are dependent upon the precision with which the underlying phenomena are understood. Even if real image sequences cannot be reduced quite to the simplicity and compactness of computer animation representations, a more flexible representation of the objects in the scene will still permit many computer-graphics-style interactions and variations to be performed. Traditional computer graphics, of course, is concerned with manipulating synthetic objects in a synthetic environment, and creating a similar degree of control over images of real objects would open up many new creative and communicative possibilities. But the most exciting scenario involves total removal of the distinction between real and synthetic subject matter. The line between computer graphics and photography/cinematography would thus become blurred, and many hybrid applications would develop.

It must be noted that any technique useful to a moviemaker for post-production can instead be used by the viewer – the developments described herein aim to change not only the way in which movies are made, but also the way in which they are watched. For instance, a shopper might take a new sofa from an electronic catalogue stored on a compact disc and view it placed into a variable-viewpoint picture of his or her own living room (real object into real scene). An architect could view in the context of its eventual site a building designed using a computer-aided-design system (synthetic object into real scene). Or a moviemaker could have human actors perform in a computer-graphics

set (real object into synthetic scene). The desirability of the latter possibility has for a number of years been recognized by Hollywood cinematographers; Mathias and Patterson specifically address this point in their book on cinematography:

> "It is entirely conceivable, for example, that all the sets and locations for a movie could be created electronically... The tools available for electronic cinematography today may still be crude in comparison to the ultimate potential, but the groundwork has been laid." [Mathias, 1985](p. 237)

Synthetic-object/synthetic-scene graphics would also benefit from stronger functional ties to imaging of real objects – both because of the understanding of real-object representations that would develop in such a research environment, and because in some cases the "image" representation of a real object might serve as a starting point for creation of synthetic scenes.

Taking a picture with a lens involves collapsing a three-dimensional world into a flat array of intensity values, which may be imaged upon a piece of photographic film, an electronic sensor, or the retina of an eye. Such a projection necessarily involves discarding the distance coordinate of each intensity point imaged;[2] 3-D shape information is lost, and mapping scenes to 2-D images is thus a many-to-one transformation. Perspective projection further confounds distance with planar extent: far-away objects occupy less of the image plane than near ones of similar size. Any attempt at trying to recover

[2]Unless the phase of the incoming light is recorded, as in holography.

"what's actually out there" from a planar projection will involve undoing perspective effects caused by the imaging system, as well as filling out the volumetric shapes of objects. Both of these tasks require recovery of distance information for use alongside the intensity information, something that the brain does quite well. The human visual system has been shown to use a variety of distance cues, among them binocular disparity, motion parallax, texture gradient, surface shading, size perspective, occlusion, aerial perspective (atmospheric blurring with distance), focusing, and of course knowledge of object structure based upon prior experience [JGibson, 1950][Jarvis, 1983]. While much vision science and artificial intelligence research aims toward allowing a computer to interpret images as the human visual system does, this goal is far from achieved [Marr, 1982]. In order to render subsequent processing tasks more manageable, it may be necessary for the camera to measure some additional scene information directly.

A good example of a movie camera that relies on non-pictorial information to enhance the usage of the image sequences is the Aspen Movie-Map, in which the camera was linked to an odometer and took frames at precise distance intervals and directions [Mohl, 1982]. This arrangement was central to the ability to locate the frames within a geographical database, and to reassemble the frames to create an individualized viewing experience. Still, since only one positional value was available per frame, the interactions available within the Movie-Map were strictly at the frame level. The only views available to the user were exactly those taken by the original camera, and there was no

way to step outside the street grid or the front/back/left/right/aerial choice of views.[3]

More recently, Chesnais has instrumented a medical arthroscope camera to return position and heading information. The data returned is used to generate computer-graphic "signposts" for combination with the live camera video [Chesnais, 1988]. This camera may be said to be "smarter" than the Aspen camera in that the system has an internal model of what the camera is likely to encounter. But this model is static. Apart from the computer graphics image additions, the camera's video is simply passed through in analog form and not processed; the system's model does not change as a result of what the camera actually sees.

1.2 Overview

A goal of this thesis is to have the range-sensing camera be as much as possible like an ordinary film or video camera, and capable of simple operation under a variety of real-world conditions. While it may not work exactly like the eye-brain combination, it should – like biological vision systems – incorporate several different sources of range information and over time build up as complete as possible a model of the objects it sees. The camera and associated processing system will have no *a priori* knowledge of scene contents, and will not recognize specific objects, but will make reasonable,

[3] A more interesting extension was explored in which frames were taken with a Volpi lens, which gives panoramic views encompassing 360 degrees of rotation, and these images anamorphically processed to synthesize any possible view direction [Yelick, 1980]. The analog manner in which Aspen frames were stored limited the processing options, however, and the viewer's position was still restricted to that of the actual camera.

general assumptions about lighting, reflectivity, optics, smoothness of object surfaces, and dynamics.

The second chapter of this thesis discusses and demonstrates some of the unique manipulations possible upon real scenes when range data is available to supplement an ordinary camera image, and proposes a database format combining both range and intensity information.

Chapter three examines various passive-camera techniques of rangefinding, and develops in detail and analyzes the method of depth-from-focus.

Chapter four describes a method for extracting the motion of rigid objects from a sequence of input images in conjunction with depth-from-focus, and – by virtue of the overconstrained nature of the problem – using motion to refine the range information as well.

Chapter five shows how a relatively complete model of scene objects may be built up through time when range and motion are computed using the techniques developed in the preceding chapters.

Chapter six considers the lessons learned through this thesis project, and proposes further directions this work might take.

This document covers a significant amount of territory in the realms of computer graphics, optics, dynamics, machine vision, and digital signal processing, and it has been necessary to limit detailed technical discussion of common concepts and techniques that are adequately described elsewhere in the literature pertaining to these fields. As

readers may not be conversant with all these areas, an attempt has been made to provide background references which it is hoped will provide sufficient additional information.

Chapter 2

Combining intensity and range

who dreamt and made incarnate gaps in Time & Space through images juxtaposed, and trapped the archangel of the soul between 2 visual images and joined the elemental verbs and set the noun and dash of consciousness together jumping with sensation...

Allan Ginsberg

"Howl"

2.1 Active rangefinding

In industrial and robotic applications biological vision is rarely mimicked. Commercial machine vision systems which require measurements of distance have for a number of years relied upon specialized, active range sensing hardware. The greatest success has been enjoyed by laser rangefinding techniques, in which a spot or pattern of light is

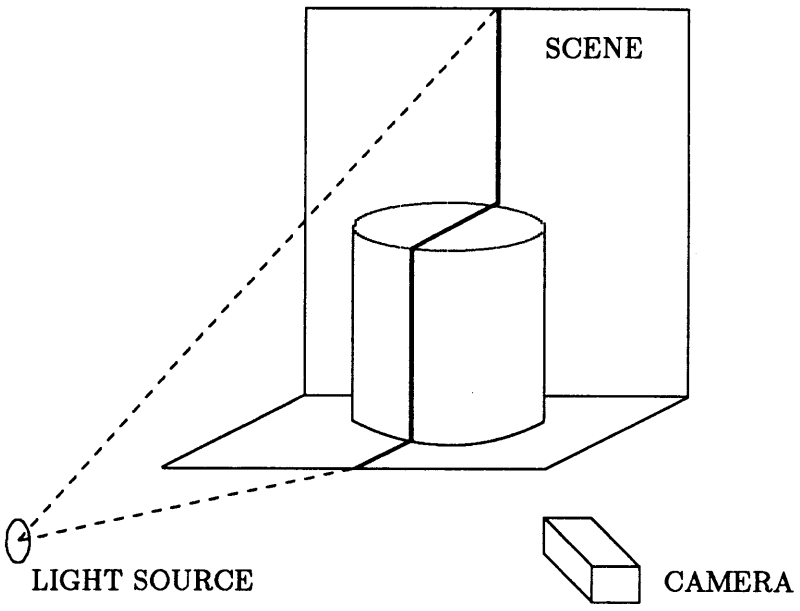

Figure 2-1: A light-stripe rangefinder.

scanned across the scene and either triangulation [Agin, 1973] or time-of-flight [Nitzan, 1977] is used to compute distance to points from which the light reflects.

As a beginning part of the investigation into the use of range data describing real scenes, a scanning light-stripe rangefinder was built. The light source consists of a small semiconductor laser whose output is passed through a cylindrical lens to create a vertical stripe of light, and reflected from a rotating mirror attached to a galvanometer whose position is controlled by the computer. From the camera's viewpoint, this stripe undergoes a shift to the left which increases as the surface it hits nears the camera (Figures 2-1, 2-2).

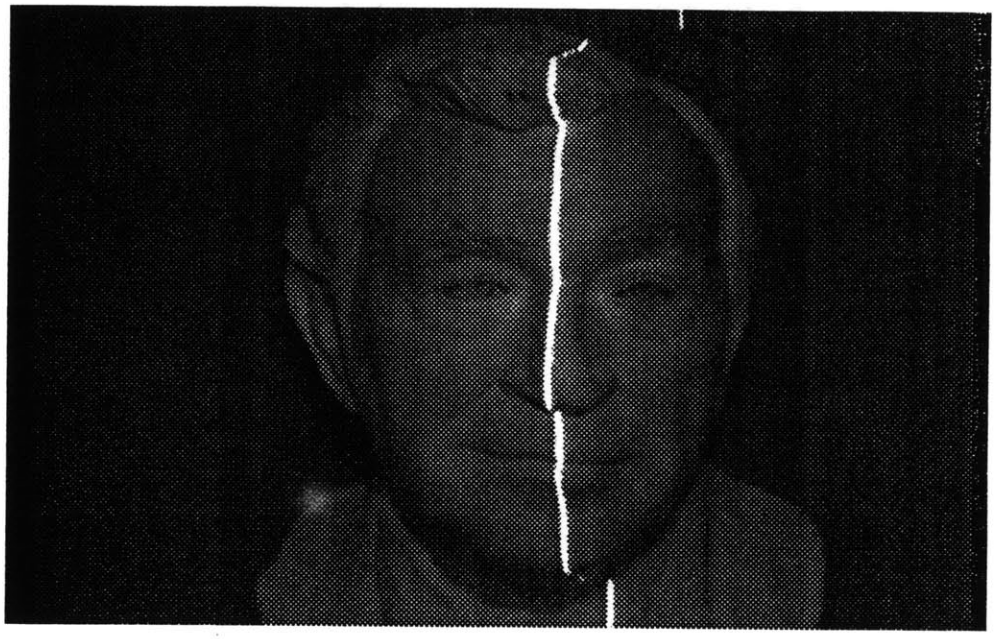

Figure 2-2: View seen by the camera in a light-stripe rangefinder.

The trigonometry involved in finding distance given the position at which the beam is placed and that at which the camera sees it is illustrated in Figure 2-3. Here the system has been set up in a manner which simplifies calculation – the mirror and camera have been placed at the left and right edges of the scene, respectively, and the outside edges of their fields of view form right angles with the baseline between them. Consider the case of some scene point **P** hit by the beam. The angle θ_{mirror} is known, as the mirror's position is controlled by the computer, and θ_{camera} is easily found from the position of the beam in the frame buffer, since each pixel may be thought of as

subtending a fraction of the lens' (known) field of view. Clearly,

$$\tan\theta_{mirror} = \frac{a}{c}, \quad \tan\theta_{camera} = \frac{b}{c}. \tag{2.1}$$

Here a and b are not known, but their sum is a known constant. Combining and rearranging,

$$c = \frac{a+b}{\tan\theta_{mirror} + \tan\theta_{camera}}. \tag{2.2}$$

If the distance from the camera, not the perpendicular baseline distance, is desired,

$$z = \sqrt{c(c + \tan\theta_{camera})}. \tag{2.3}$$

A triangulation-based rangefinder of this sort will characteristically suffer occlusion problems, in that camera-visible points on the extreme right-hand sides of objects may fail to be illuminated by the laser; also foreground objects cast shadows to the right onto surfaces behind them, cutting off the light source.[1] These occluded regions have indeterminate range. Designing this type of system involves a tradeoff: a larger light-to-camera offset increases accuracy (since a given change in distance will result in a larger shift of the beam), but at the expense of larger occluded regions. The rangefinder described here attempts to minimize the problem by extrapolating the surface at the right of the occlusion into the "hole" – a process which works reasonably well on most

[1] It is also possible to scan the scene twice, using light sources on both sides of the camera.

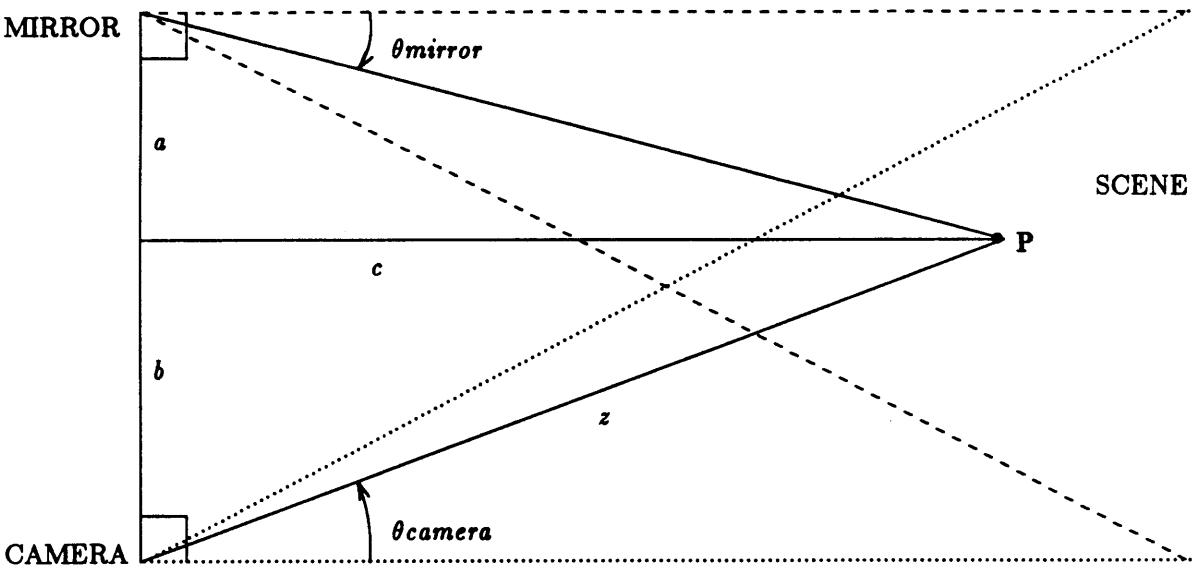

Figure 2-3: Triangulation is used to solve for distance z from the camera.

of the "shadow" occlusions described above but does not produce quite correct results on the visible-but-not-illuminated occlusions. Generally, the laser range camera is set up such that the camera-to-scene distance is much greater than the scene width, and the occlusions are small enough that the simple hole-filling scheme is adequate for the applications to be described below.

2.2 Representation of three-dimensional scenes

Marr called a representation that includes intensity and range information from one viewpoint (such as that provided directly by a rangefinding camera, or as derived by the

brain from binocular images) a "2 1/2-D sketch," and differentiated it from a three-dimensional model in that it is viewer-centered and contains only one z value for a particular x and y [Marr, 1982].[2] Unlike a three-dimensional representation of a scene, it appears continuous from only one viewpoint. Consider a cylindrical object: as it curves away from the viewer its sides foreshorten, and equal areas on its surface are represented by fewer and fewer range values on the range image's "surface," until its back can't be seen at all. In effect, the range reading for each point represents the first surface hit by each line of sight outward from the camera. When the information representing the cylinder is fed through computer-graphic transformations to render it rotated 45 degrees about its axis, two significant phenomena emerge. First, the points which represented the foreshortened sides (and are now in the "front" relative to the new viewpoint) are too sparse to present a continuous surface. Also, what was the back of the cylinder relative to the camera is not represented at all, and if the object was originally in front of a wall a rectangular hole is now visible therein.

Thus the initial 2 1/2-D sketch will be inadequate as a scene representation. The sparseness problem mentioned above presents little trouble, for if local smoothness can be assumed new points are easily interpolated, or a rougher surface's statistics can be calculated and new points generated via the statistical model [Pentland, 1987a]. Reconstructing the back faces of objects and occluded portions of the background will

[2]The question of exactly how distance is represented in biological vision systems is still open. Marr discusses the possibility of a coarsely-quantized sense of absolute distance, along with a finer representation of local differences in depth (*i.e.* surface orientation). These different representations presumably obtain their data from different processes, though they would perhaps be checked for consistency.

require additional information from another source.

It is possible to combine the views from several cameras to provide complete representations of object surfaces if information about the camera positions is known [Linhardt, 1988]. This process is more complicated than might first appear. Uncertainty in measurement of camera location and orientation, as well as range data distortions that may vary from camera to camera preclude combining the range points in a simple manner; a data-driven correspondence process is necessary so that the points from different views coalesce to form coherent, undistorted objects. Many events and programs are shot with more than one camera, and were these cameras capable of sensing range it would be possible to synthesize images from intermediate camera positions.

When only one camera is used, reconstruction of occluded regions is possible, given that they have been seen at some earlier point in the image sequence[3] and given that the motion of objects in the scene can be calculated. Reasonable assumptions must also be made about objects, such as that they are totally rigid, or at least articulated from rigid parts – this is so that the points that are not visible may be assumed to be moving in a manner consistent with those seen by the camera. If this path is pursued, there is always the possibility that the entire surface of an object is *never* seen throughout a sequence. In this event, one of a number of assumptions may be in order: the software may simply render the points the camera has seen, and leave it at that; if only a

[3]If the entire sequence is batch-processed, then the points merely have to be visible *somewhere* in the sequence.

small gap exists in a surface description then points may be filled in from the edges; if the object seems generally convex then rotational or front-to-back symmetry may be assumed. The choice of assumption might be made on an object-by-object basis based upon overall shape or local surface characteristics.

Irrespective of the number of camera views combined, or the amount of time that moving objects are tracked, there will always be some portions of the scene for which there are insufficient data to perform a complete reconstruction – the inside of a hollow object with a small opening, for example. The point of this process is not to enable the synthesis of absolutely any camera position, but rather to permit relatively accurate rendering of a scene from a large number of useful viewpoints other than those of the original camera or cameras.

Regardless of their source, once additional points have been generated for solving the sparseness and occlusion problems, the data no longer fit into the original range-image/intensity-image database. Now there will be more than one z and intensity value for many values of x and y, necessitating something other than a two-dimensional array. The three-dimensional analogue of pixels are called "voxels," or volume elements, which span three-space.[4] Converting a 2 1/2-D image representation of an N-by-N-pixel image to voxels will increase the array size from $2N^2$ to N^3, though the vast majority of voxel cells will be unoccupied, as they represent empty space. A storage optimization

[4]Some writers reserve the term "voxels" for true volumetric data, where the interior of an object is filled, and refer to rangefinder data, which really only represents the outer surfaces of objects, as "surfels".

sometimes applied is the "oct-tree," in which a cubic volume is divided into octants, each of which is further divided into octants, and so on. This space is represented as an 8-ary tree in which each node has either a label which serves to describe all its sub-octants (full or empty), or eight leaves [Jackins, 1980]. Methods for efficient computer graphics translation, rotation, scaling, and other operations have been developed for use on objects stored in oct-trees [Meagher, 1982].

Rather than such a volume description, synthetic computer graphics rendering software often applies surface representations, in which a particular (x, y, z) point cannot be read directly from the database but must be calculated. A popular description represents surfaces as assemblages of polygonal facets, recording the locations of the vertices. More accurate and efficient representation of smoothly curved surfaces is provided by other means, among them parametric cubic surfaces [Foley, 1982][Mortenson, 1985], and superquadrics [Pentland, 1987a].

When voxels are stored in a list format rather than in a 3-D array, they are commonly called particles, and attracted the interest of computer graphics researchers who wished to model natural objects and phenomena difficult to describe via a polygonal representation [Crow, 1982][Reeves, 1983]. Rendering systems are in use which allow both types of representations to co-exist. Particle representations as used by Reeves are created and manipulated by stochastic processes, and as such can be operated so as to provide "data amplification,"[5] where additional particles can be automatically

[5]This term apparently was first used in a paper by Alvy Ray Smith [ASmith, 1984].

generated to increase the density of an already-existing set [Reeves, 1985]. The process is simply a matter of interpolating new particles based on the statistics within the vicinity of a point in three-space. This property seems to offer a solution to the range data sparseness problem as described above, as well as permitting a real scene representation to be "upconverted" for rendering on a display having higher resolution than the camera system which provided the original data points. The stochastic property implies the ability to upconvert temporally as well, provided the scene is temporally sampled at a sufficiently high rate to avoid significant aliasing.

A simple and rapid, if memory-inefficient, method for increasing particle density is to unpack the particle list into a 3-D array, and to generate new particles in empty cells with a probability based upon the density of particles in surrounding cells. The color of a new particle is determined by the colors of neighboring particles. When a large number of new particles is generated in this manner (or when the process is performed repeatedly), however, surfaces tend to become "fuzzy." A more satisfactory result is obtained through applying the assumption of local surface smoothness: within a volume of three-space, the least-squared-error method is used to fit a planar, biquadratic, or higher-order surface to already-existing particles, and new ones are generated in empty cells that lie on this surface.

In the case of real images, going from the 2 1/2-D representation to particles is a simple process. Each particle in the list has x, y, and z values, a color, and perhaps

other context-specific attributes such as a surface normal vector,[6] a description of its specular and diffuse reflectivity characteristics, or a label describing the object to which it belongs (the ways of generating this will be discussed in the next section). Manipulation and rendering of the data will be made more efficient if a requirement is placed that all particle-manipulation software produces object descriptions sorted in scan and depth order, that is sorted by y, within y by x, and within x by z.

Two other important operations must be performed upon real scene data when it is placed into a particle database. The first is perspective removal, or expanding distant objects so that their (x, y) extent is not dependent upon their distance. The expansion factor with distance will be determined by the view angle of the camera which provided the original images, which is a function of the focal length of the lens employed. Expansion of distant objects will cause the particles composing them to be sparser throughout space, and provides an opportunity for the data amplification feature of particle databases to be exploited.

If it will be desired to modify the lighting of a scene, a common rendering operation from computer graphics, the unit surface normal vector to each point in the original range image should be calculated by differentiating the range information, and the components of these normal vectors stored with each data point in the particle representation (new particles created by interpolation can have their normal vectors

[6]The surface normal will, of course, be calculated on the surface of the original input range image. Here, clearly, the particles are acting as surfels rather than voxels, inasmuch as a voxel has no surface orientation.

interpolated as well). Finite differencing of range values across x and y can provide the corresponding components of the surface normal vector, while the z component comes from the fact that the vector is constrained to have unity magnitude. Since, however, the range values from the laser rangefinder are given as integers, one-pixel differencing will produce undesirable results on gently sloping surfaces, where the normal vector will alternately point straight at the camera and then be sharply angled at the "steps" in the range value (Figure 2-4).[7] The solution is to apply not a unit doublet (the first difference of a unit impulse) as the differentiation operator, but rather the convolution of a unit doublet with a smoothing filter such as a Gaussian. Legitimate sharp edges in range can be preserved by making the filter adaptive, *i.e.* the filter window shrinks on one side or the other so that it does not extend across large discontinuities in range.

Further, the "color" of each particle should really represent its reflectivity rather than the product of its reflectivity and the ambient illumination. In a studio setting, this requirement suggests the use of very diffuse, or "flat" lighting; in other cases the surface normals calculated from the range data may be employed with some success in un-doing the effects of strongly directional lighting if an estimate is available of the position of the light source. This will involve running in reverse the lighting model described in the next section.

[7] Another way of stating this problem is that the range information is not bandlimited; the "steps" are the manifestation of unwanted high frequencies.

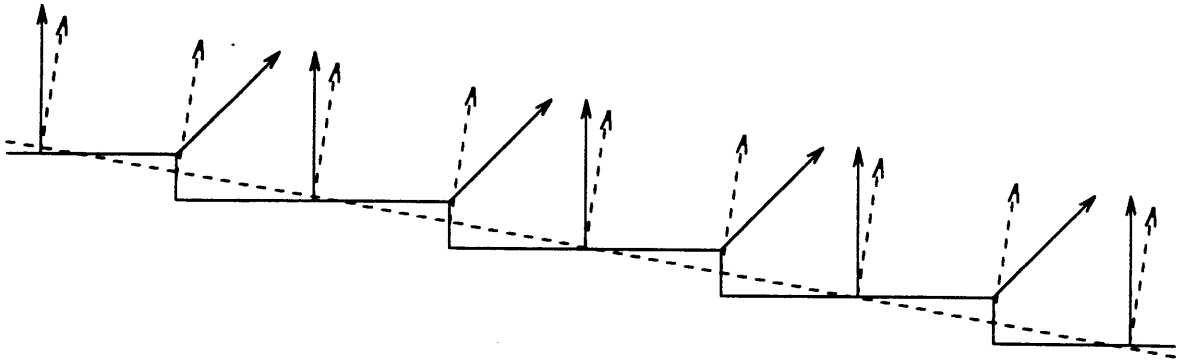

Figure 2-4: Computing surface normals on a coarsely quantized surface via one-pixel differencing will produce incorrect results (solid arrows). Dashed arrows show the desired result.

2.3 Applications of range information in image display

Given a particle description of a real scene, rendering from some viewpoint other than that of the original camera is a matter of feeding the particles' coordinates through a set of matrices which perform the necessary rotations, translations, and perspective transformations [Newman, 1979]. The perspective transformation applied in rendering need not be to the same view angle as the lens which originally imaged the scene, and therefore the apparent focal length may be modified at this step (Figures 2-5, 2-6, 2-7).

Varying the lighting of a computer graphics scene is often done to set a particular "mood," or to best illustrate surface features of objects. Real scenes have their lighting

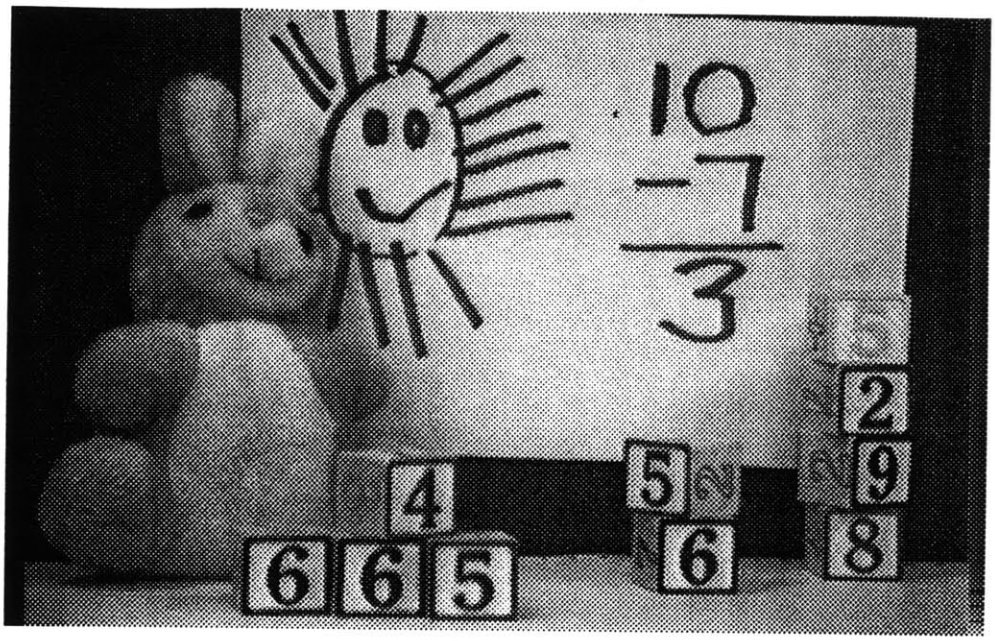

Figure 2-5: Intensity image of a scene.

arrangements set up for similar reasons; but it may be faster and easier to modify the lighting with a computer than by moving physical light sources around. It also may be desirable to make quick changes in the lighting with a computer for illustrative purposes (to see, for example, how a scene will look at different times of day) or simply because it was not physically possible to set up lights in the desired locations while the scene was imaged. In a commonly used illumination model, the intensity with which a particular point is rendered is a function of the normal vector to the surface at that point, the locations, colors, and and intensities of all light sources incident at that point, and the reflectivity characteristics of the point, which are broken into specular

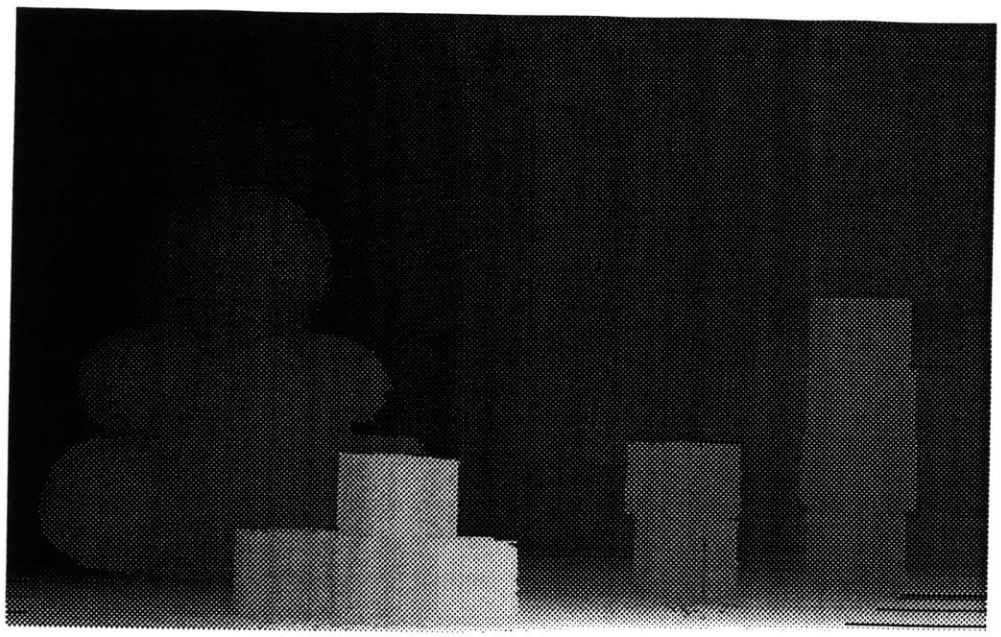

Figure 2-6: Range information for the scene in Figure 2-5 represented as a "range picture," where closer points are lighter.

and diffuse components:

$$I = I_{ambient} k_{ambient} + \frac{I_{source}}{r^2}[k_{diffuse}(\mathbf{l} \cdot \mathbf{n}) + k_{specular}(\mathbf{r} \cdot \mathbf{v})^n] \tag{2.4}$$

where $\mathbf{l} \cdot \mathbf{n}$ is the cosine of the angle between the unit surface normal and the unit vector to the light source, and $\mathbf{r} \cdot \mathbf{v}$ is the cosine of the angle between the unit vector in the direction of reflection and the unit vector toward the viewpoint [Foley, 1982]. This is a strictly empirical approximation, though realistic results can be obtained with appropriately selected values for the proportionality constants k and for the exponent

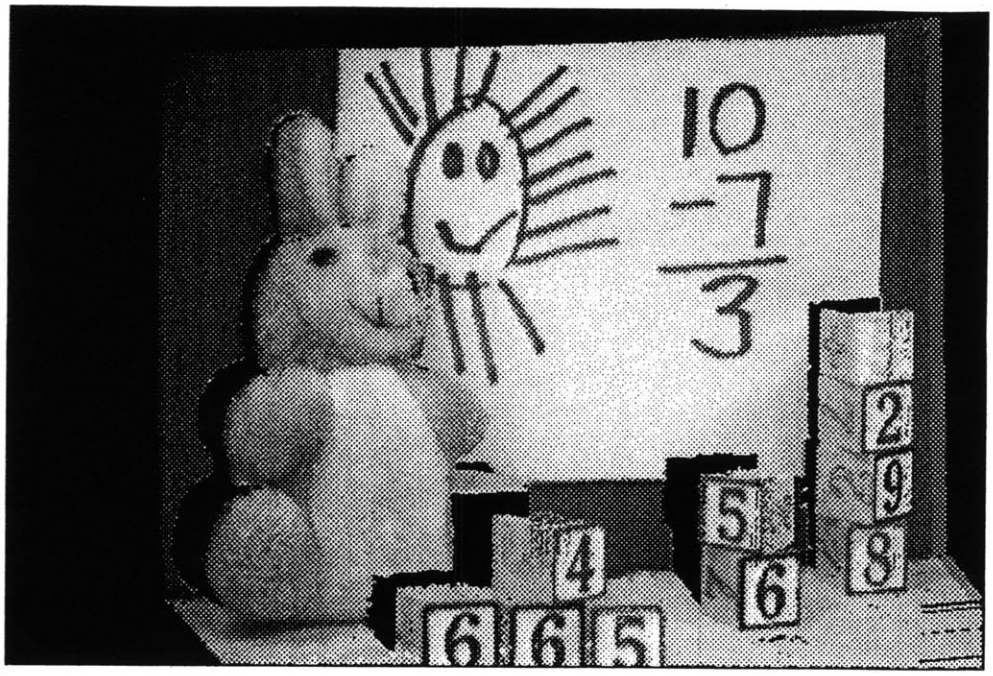

Figure 2-7: The data from Figures 2-5 and 2-6 rendered from a different viewpoint. Note the "holes" resulting from the 2 1/2-D nature of the information.

n. For a color image this model must be applied three times to the red, green, and blue components of the image, and for multiple light sources the contributions from each source must be summed at each point. In Figures 2-8 and 2-9, purely diffuse reflectivity is assumed; better renderings of real scenes would be available if the camera had some way of estimating the actual reflectivity characteristics of points in an image. Sensing reflectivity would require having multiple views of a scene illuminated from different, known directions (see [Horn, 1986]). The simple illumination model presented here permits rapid image calculation, but proper rendering of shadows and interreflections requires more sophisticated ray-tracing techniques [Cook, 1984], in which directed rays

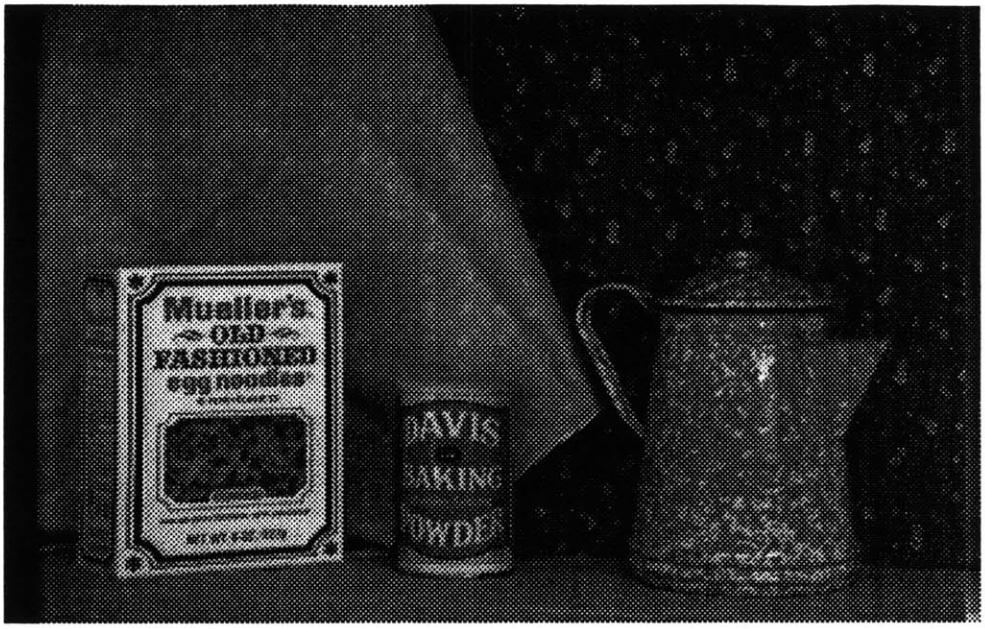

Figure 2-8: A scene with fairly diffuse illumination.

of light are followed from the light source toward object surfaces in the scene and the final intensity of each point is the sum of the contributions of each ray incident thereupon.

A gradient of focus creates a strong sense of depth in an image, and is an excellent way of drawing a viewer's attention to one region or object in a complicated arrangement of objects. Photographers and moviemakers (and even painters) have long taken advantage of this knowledge, though the possibilities for applying focus to interactive information systems have apparently been little explored. Potmesil and Chakravarty, as well as others, have investigated applying realistic simulated defocusing to computer

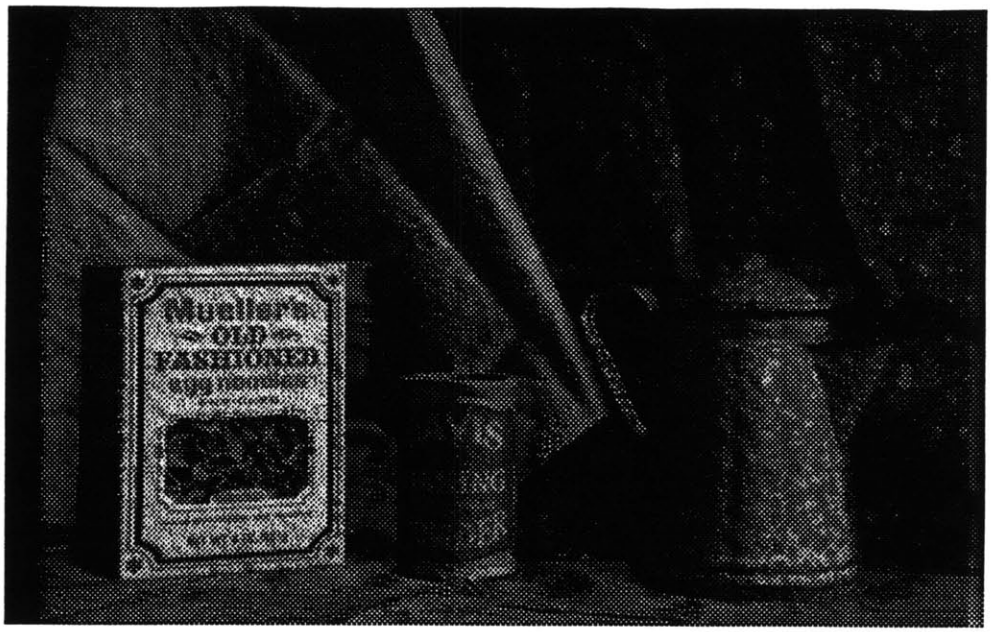

Figure 2-9: The scene of Figure 2-8 with a computer-simulated light source on the right.

graphics [Potmesil, 1981][Cook, 1984]. The mathematical details of focusing will be explored in the beginning of the following chapter. At this point it will suffice to observe that if a scene is imaged with all regions in focus, then knowledge of range for each pixel permits synthesizing the defocusing effect of a lens with a larger iris opening (Figure 2-10).

The combination of elements from more than one source to create a scene is a process known to filmmakers as matting and to television producers as keying. Keying is a process by which a "hole" is cut into a video image and another image is shown through the cut-away region. Perhaps the most common method is chroma-keying, which is used

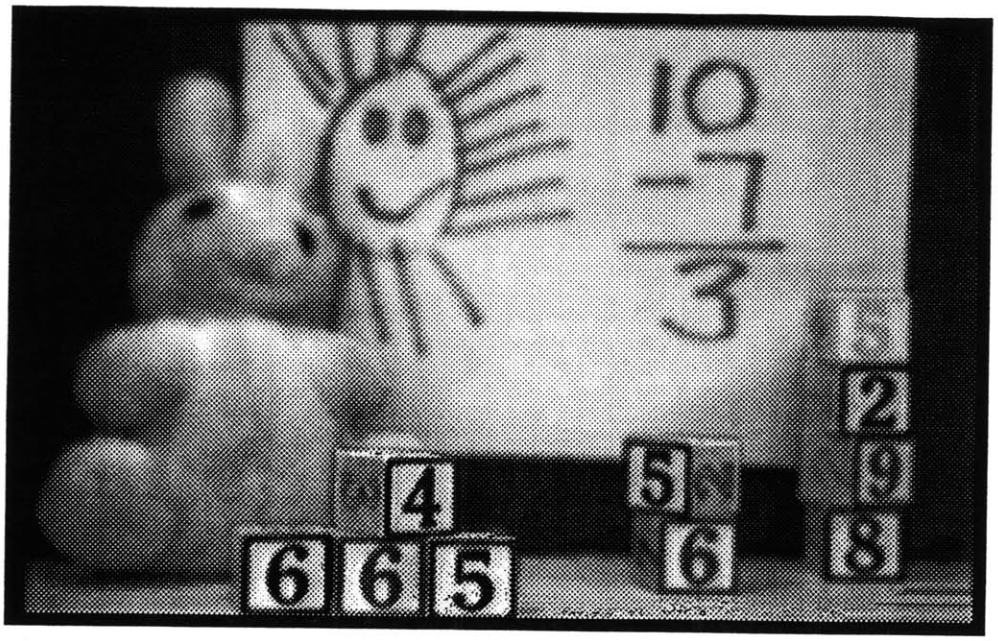

Figure 2-10: The scene of Figure 2-5 with a computer-simulated lens focused on the front row of blocks.

by local television stations to provide a changeable background for the weatherman. In the studio, the weatherman stands before a very saturated blue (usually) wall, and the keying hardware simply cuts the hole wherever it detects bright blue. The process is quite effective, though the announcer cannot wear a bright blue necktie lest a necktie-shaped piece of the background appear in the center of his shirt. A similar technique, called bluescreen matting, is used in film, though some cinematic matting continues to be done via frame-by-frame hand drawing of mattes or photography of miniature models.

Such keying/matting methods are insufficient for many purposes, not only because

they require special preparation of backgrounds, but also because the process is analogous to cutting out a piece of one photograph and pasting it on top of another. Occlusions are not handled properly, in that a person placed into a scene by matting cannot walk behind elements of the scene into which he is placed (because, indeed, he is not placed "into" the scene at all – merely *onto* it). When range information is added to the process and all scene elements – foreground and background, actor and set – are passed through computer-graphics-style rendering, the occlusions will work correctly. At this point the process is no longer really keying (though a user may continue to think of it as such) but more properly the assembly of scene elements in a three-dimensional space (Figure 2-11).

An early mention of using distance for keying purposes appears in a 1981 patent [Rose, 1981], for making a longer effective depth of field than is possible with lenses typically used for video production. Rose's idea is to take an image from a lens focused at a distance and another from a lens focused on the foreground, and to combine the sharpest regions (by which Rose means those having more high-frequency content in the video signal) of each. No mention is made of the fact that as a lens changes its focus position its magnification changes significantly, nor does the system proposed appear to compensate.

When it is desired to pick out a particular object from a scene (and even more in a later chapter, when the three-dimensional motion of individual objects will be tracked), the input images will have to be segmented into regions representing in-

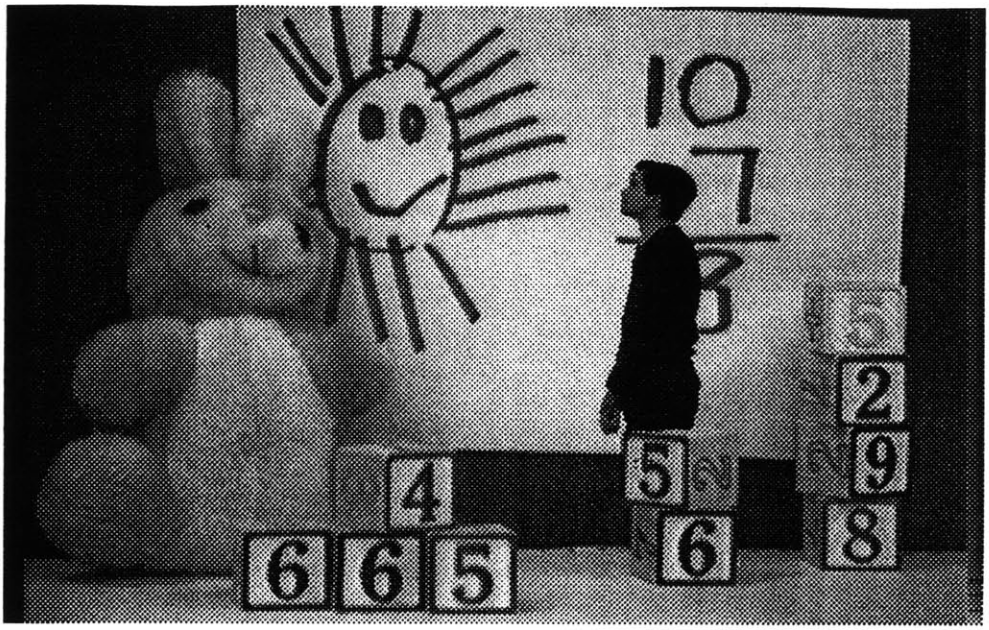

Figure 2-11: Using range information to generate correct occlusions when combining elements from different scenes.

dividual objects. Segmenting intensity images is a difficult process [Fu, 1987], and the introduction of range information, while providing an independent, robust, and illumination-insensitive set of cues, does not (as was apparently hoped by some early artificial intelligence researchers) automatically solve the problem. Range images are in general so much "better-behaved" than intensity images that when rangefinding equipment first became available the corresponding intensity images were essentially ignored and the problem of scene segmentation changed to one of segmenting only range images. Various scene-segmentation methods have been applied to rangefinder data, from drawing object boundaries at large jumps in distance [Nitzan, 1977] to calculating

surface orientation and curvature and classifying objects based upon these characteristics [Inokuchi, 1983][DSmith, 1985][Archibald, 1986]. Less-investigated but presumably more promising is the process of looking for object boundaries in both range and intensity images of the same scene [Gil, 1983]. Rough surfaces will not necessarily respond well to such simple schemes, but it has been suggested that they be modeled as fractal proceses, and that they be segmented based upon their surface statistics [Pentland, 1987a]. The more elaborate schemes take advantage of the typical high quality of active rangefinder images, which commonly provide spatial resolution nearly equal to that of an intensity image from an ordinary video camera. Passive ranging methods (as described in the next chapter) will provide smoother surfaces with less x, y, and z resolution, so texture-based segmentation is not as likely to work here, but Pentland also notes that the fractal dimension of the intensity function is in many cases related in a known way to the fractal dimension of the three-dimensional surface, and therefore fractal segmentation of the intensity image might be employed to confirm the object edges produced through other processing of the intensity and range images. Such segmentation can be done by computing the ratios of power in several one-octave-wide bands in the frequency domain, constructing a histogram of the ratios across the image, and adaptively quantizing the resulting distribution.

The image segmentation issue is clouded by the fact that there is no generally-agreed-upon notion of just what defines an object. The method of image segmentation that has been employed in this project is somewhat empirical in nature, but provides

relatively good segmentations of scenes in which objects do not interocclude in complicated ways (beyond some level of complexity, it probably becomes almost mandatory to add knowledge of object models to the process). A large change in z is almost certainly an object boundary, while a smaller change might be due to an object boundary but might simply be the result of surface texture or range data noise. The likelihood that such a "weak" edge defines an object is increased in the presence at the same location of an edge in intensity, in hue (if a color image is available), in surface orientation, or in texture. In logical terms, the decision is made to construct a boundary through points at which there is a

```
((large change in depth) or
 ((smaller change in depth) and
  ((large change in intensity) or
   (large change in hue) or
   (large change in surface normal) or
   (large change in texture))))
```

where parameters like intensity are averaged over a small window to increase the noise immunity of the algorithm. Closed contours are constructed through the points which the boundary detector labels, and the regions inside each contour are given a unique label when the enclosed points are put into particle form. An obvious improvement to this algorithm is individually weighting contributions from edge detectors operating

in various domains (intensity, depth, *et cetera*), and thresholding their sum to decide whether or not there is an object boundary. The range-intensity pair of Figures 2-5 and 2-6 is segmented in Figure 2-12. It is significant to note that the rabbit's ears have been segmented as separate from the body, apparently due to changes in intensity and surface orientation at the boundary. Also, the blocks have for the most part not been segmented one from another – which is really not a problem unless they start moving independently. In the absence of detailed internalized knowledge about objects the camera is likely to encounter, the system can really only make a likely segmentation which a human operator may wish to modify, or which may be changed by later observations of the scene. A segmentation method that would work particularly well in an interactive environment is the application of energy-minimizing splines, or "snakes" [Terzopoulos, 1987], which typically are placed by a user in the rough vicinity of a desired contour, and their affinity for a particular feature (such as an intensity edge) draws them toward the correct position.

When some region segmentation method, however good, is applied to individual frames of a movie, there is no guarantee that the same regions will be given corresponding labels in each frame, or even that the same number of segments will be identified on each frame. A technique which works better is segmentation of one frame followed by transformation of the object regions so that they match the range and intensity information of successive frames of the scene, with occasional resegmentation in the event of unresolvable problems (such as objects entering or leaving the scene).

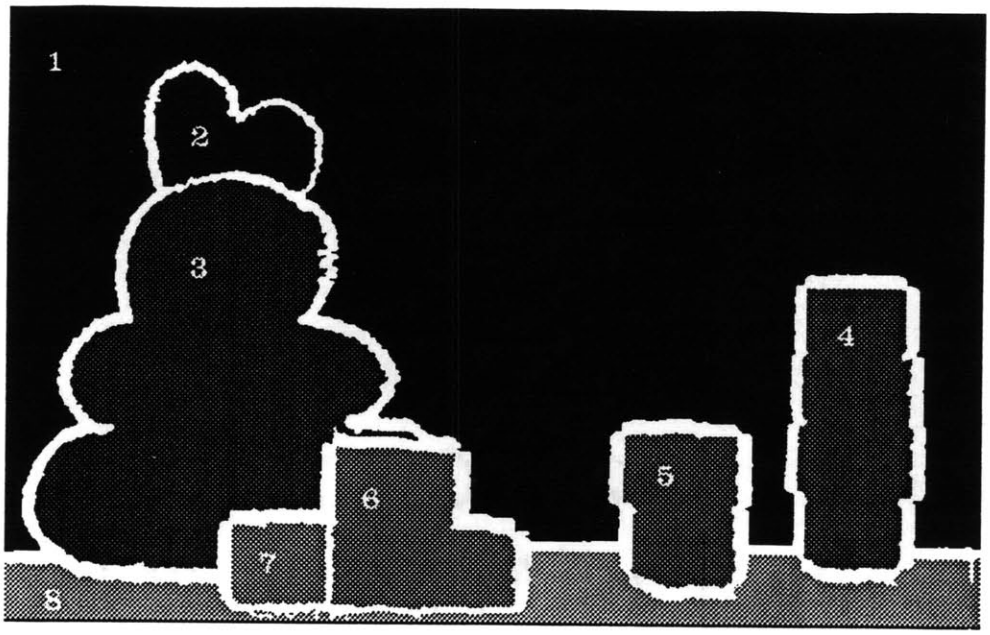

Figure 2-12: Segmentation of a scene by examination of edge cues in multiple domains.

The esthetically-controversial Colorization(R) technique, by which old black-and-white motion pictures are made into color video tapes by overlaying a synthetically-generated hue component, works as follows: a hue overlay map is drawn by an artist for the first frame of a scene, and (by use of autocorrelation for motion detection and logic which attempts to keep the hue overlays coherent) the system automatically makes the hue patches track intensity regions through the following frames of the input images. Of course, problems which cannot be resolved automatically, such as a new character entering mid-scene, require operator intervention [Markle, 1984].

An experiment was done in which Markle's simple object-label tracking method

was attempted on a sequence of moving images segmented as described above. The algorithm appears to perform quite well given both range and intensity information as input to assist in the tracking. Kass, Witkin, and Terzopoulos have shown that "snakes" may be used to follow edges from frame to frame in a movie, once they have been locked on to a feature [Kass, 1987]. But when real motion estimation information is available (as will be the case in later chapters of this thesis), the proper solution to the problem is to assume that inertia will limit the rate of change of object motion, which means that the motion detected up to the current frame may be extrapolated to the next frame time to give an estimate of where the object will have moved. This can then be checked for consistency with a re-segmentation of the next frame.

2.4 Chapter summary

A database format has been introduced for use in representing real images for which both range and intensity information are available in either 2 1/2-D or volumetric 3-D form. Using this representation it is possible to perform computer-graphics-style manipulations upon real scenes, including variable-viewpoint rendering, focus simulation, lighting variation, scene segmentation into objects, and assembly of objects from different scenes.

Chapter 3

Depth-from-focus

This is the short and the long of it.

William Shakespeare

The Merry Wives of Windsor

3.1 Passive rangefinding methods

While light-beam rangefinding hardware can be very rapid, in general it is too slow to use on moving images[1] and current methods are somewhat sensitive to the reflectivity of imaged objects and the presence of large amounts of ambient light. The need for high-powered light sources to overcome ambient illumination restricts the use of these methods to relatively small, indoor scenes. These considerations suggest an investiga-

[1]Boyer and Kak have demonstrated a non-scanning technique in which a single colored pattern projected onto a scene provides enough information to derive range values for all visible, non-occluded points, thus potentially allowing range to be measured at video rates [Boyer, 1987]. The rapid flashing of such a pattern by, *e.g.* a xenon flash tube may be envisioned.

tion of rangefinding techniques which inherently offer greater speed and which do not require the camera to project energy onto the scene.

Virtually any of the distance cues exploited by biological vision should be possible for computational vision, and ideally (as none of them works well for all imagery or all viewing conditions) range should be calculated via integration of constraint data from multiple processes, as has been noted by Terzopoulos [Terzopoulos, 1986]. The shape-from-shading technique [Horn, 1986], for example, which within a smooth region of constant reflectivity can estimate relative changes in depth of a surface (but not its absolute distance from the camera), is not too useful as a primary method of rangefinding but could be used to refine distance estimates from another source. Structure-from-motion, which will be discussed in a later chapter, is (obviously) applicable only for moving objects, but again could complement some other method.

The passive-camera technique which has been the most thoroughly investigated is stereopsis, in which distance is measured by evaluating binocular disparity between two views of a scene. Disparity is the phenomenon by which the images of points at different distances shift by different amounts as a camera moves perpendicularly to the distance axis. In order to measure disparity, it is necessary to establish a correspondence between points in each view, after which the distance is found through mathematics identical to those employed in triangulation-based laser rangefinders. The task of assigning a correspondence between views is made difficult at uniform regions where unique matching cannot be done, as well as at occluded regions where some

part of the scene appears in only one of the cameras' views (the "missing parts" problem) [Levine, 1973][Yakimovsky, 1978]. Problems correlating intensity patches between views led to feature-based stereopsis techniques, most notably the Marr-Poggio algorithm [Marr, 1982]. Marr and Poggio propose matching zero-crossings of bandpass copies of the images, based upon the observation that these zero-crossings correspond to physically significant phenomena such as changes in reflectance, surface orientation, or distance, and also based upon evidence that biological systems perform various bandpass operations as part of low-level vision.

The sparse range data resulting from feature-based stereopsis must be subjected to an interpolation operation to construct surfaces across the gaps between the features that are matched. Grimson has proposed applying a smoothness constraint in order to construct a plausible surface that could have given rise to the detected zero-crossings [Grimson, 1981]. Grimson suggests that the "best" surface is that which minimizes the quadratic variation in the gradient of the surface:

$$\Theta(z) = \sqrt{\int\int \left(\frac{\partial^2 z}{\partial x^2} + \frac{\partial^2 z}{\partial x \partial y} + \frac{\partial^2 z}{\partial y^2} \right) dx dy}. \tag{3.1}$$

Clearly any surface which minimizes $\Theta(z)$ also minimizes $\Theta^2(z)$, and a discrete approximation to this functional is what is ultimately minimized by an iterative process. This computation assumes that the z values known at the beginning of the process are exact, and provides values only for the "holes" in the range image. If, on the other

hand, the original z values are noisy, then a smoother surface can be constructed by minimizing

$$\Theta(z) = \int\int \left(\frac{\partial^2 z}{\partial x^2} + \frac{\partial^2 z}{\partial x \partial y} + \frac{\partial^2 z}{\partial y^2} \right) dx dy + \beta \sum (z(x,y) - z_{original}(x,y))^2, \qquad (3.2)$$

where β is a factor which can be adjusted to balance between fidelity to original points and surface smoothness, depending upon the known degree of error in the input points. The summation is carried out over all points for which the rangefinding process provided estimates for z. Similar minimization processes to those used to construct these surfaces have more recently been used by Gennert to develop minimum-error affine transformations for direct intensity-based matching [Gennert, 1986a]. Harris has proposed a method which reconstructs surfaces by employing sparse constraints on both depth and orientation, as might be provided by a combination of several different passive rangefinding processes [Harris, 1986].

The farther apart the two viewing positions used for stereopsis, the more accurate the resulting distance calculations. At the same time, though, the occluded regions increase in size, and correspondence between views requires searching over larger picture areas – thus increasing computational requirements.

3.2 Depth-of-field

Depth-from-focus is a passive-camera method for measuring distance which eliminates two of the major problems of stereopsis, namely correspondence and missing parts.[2] It has the additional virtue of being almost completely a signal-estimation problem, with no significant reliance on machine intelligence, and thus could feasibly be implemented entirely in hardware. The basic idea is that a lens and aperture used to image a scene will result in a gradient of focus; points will be increasingly blurred as they move away from the distance at which the lens is focused. Photographers refer to the distance range over which a scene is in sharp focus as depth-of-field, and the larger the lens aperture the shorter the depth-of-field (the more rapid the increase in blur with distance). Mathematically, this process may be modeled as the spreading of energy from image points into overlapping regions, called "circles of confusion" in photography.[3]

Circles of confusion *per se* are rarely observed in normal vision, and their discovery followed the development of the camera. The camera obscura had been in use as an artist's aid for perhaps five hundred years before the invention of photography – indeed, the image-forming properties of a small aperture were discussed by Aristotle in his *Problemata* [Ross, 1927] – but the pinholes used in the early devices had essentially infinite depth-of-field and did not exhibit the focus gradient effect. It was not until

[2] Actually, there is a certain "missing parts" phenomenon associated with depth-from-focus, but it is relatively insignificant in comparison with that from typical implementations of stereopsis. See section 3.7.

[3] This is not to be confused with the circle of *least* confusion, a term used in optics to describe the position at which a lens exhibits minimal aberration.

the sixteenth and seventeenth centuries, when lenses were used to replace pinholes to increase the light-gathering power of the camera obscura, that focusing and circles of confusion were noted [Hammond, 1981]. Circles of confusion may clearly be seen in certain paintings by Johannes Vermeer from the 1600's, particularly *Maidservant Pouring Milk* (c. 1658-1660) and *View of Delft* (c. 1660-1661). In the latter, specular highlights in the distance dissolve into diffuse discs of white paint (Figures 3-1, 3-2). *Girl with a Red Hat* (c. 1666-1667) (Figure 3-3) exhibits a gradient of blur consistent with a lens focused on the wall behind the subject. It has been a matter of dispute among art historians whether these works were literally recording camera obscura images, or whether they were painted in a style imitating optical effects which were apparently fairly well known by that period [Wheelock, 1977].

3.3 Geometrical approximation

An approximation which, though not strictly correct, is useful for understanding how defocusing varies with distance is to neglect diffraction effects. Derivation of the circle of confusion then becomes a straightforward geometrical problem.[4] The lens equation says that the reciprocal of the object-to-lens distance and the reciprocal of the lens-to-focus distance sum to a constant, which is the reciprocal of the focal length of the

[4]See [Boucher, 1968] for a different geometrical approach from that below.

Figure 3-1: *View of Delft,* Johannes Vermeer, c. 1660-1661.

Figure 3-2: Detail from *View of Delft*.

Figure 3-3: *Girl with a Red Hat*, c. 1666-1667.

lens:

$$\frac{1}{u} + \frac{1}{v} = \frac{1}{f}. \tag{3.3}$$

In Figure 3-4, a camera lens is focused such that a point X produces a sharp point image on the film or image sensor at F. Point Y, which is farther away, focuses in front of the focal plane at point C; on the film the light energy from that point is distributed over a larger region of diameter c. Since triangles ABC and DEC are similar,

$$c = d\frac{V_x - V_y}{V_y}. \tag{3.4}$$

But lens diameter d can also be expressed in terms of the focal length f and the numerical aperture (or f-number) n:

$$d = \frac{f}{n}, \tag{3.5}$$

and since a similar geometrical construction is valid for points Y closer than X (which will focus *behind* the focal plane),

$$c = \frac{f}{nV_y}|V_y - V_x|. \tag{3.6}$$

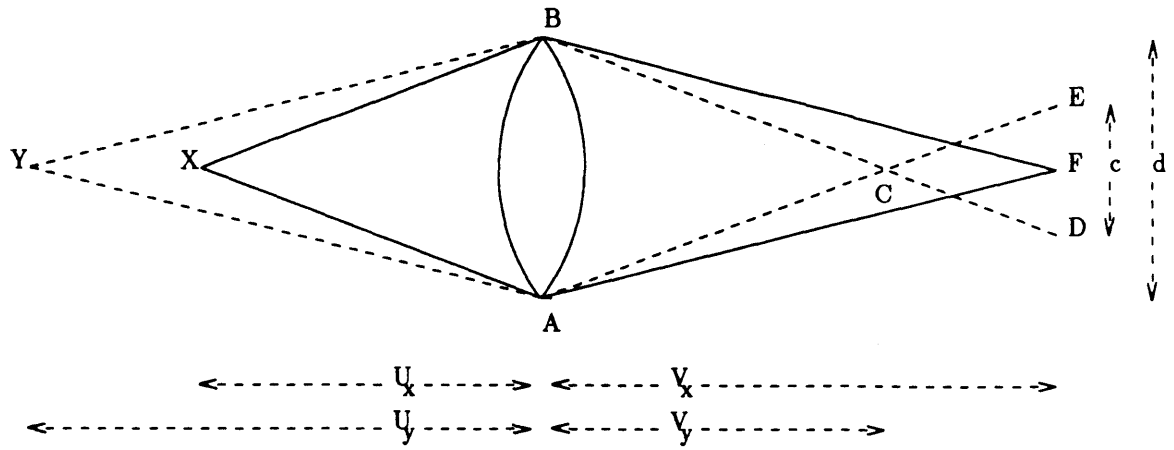

Figure 3-4: Geometrical approximation: point X focuses at F. Out-of-focus point Y images there as a blur of diameter c.

Substituting for V_x and V_y from the lens equation yields a less elegant but more useful result in terms of the object distances:

$$c = \frac{f|U_y - f|}{nU_y} \left| \frac{U_y}{U_y - f} - \frac{U_x}{U_x - f} \right|. \tag{3.7}$$

The circle-of-confusion function is plotted in Figures 3-5, 3-6, and 3-7. This function is asymmetrical about the plane of best focus. As point Y moves toward infinity, the diameter approaches a limit $(V_x/n - f/n)$, while it increases much more rapidly for points closer to the lens than point X. At $U_y = f$ the circle of confusion is the same as the effective diameter of the stopped-down lens (here the lens is behaving as a collimator). As the function rises on both sides of the plane of best focus, an inherent

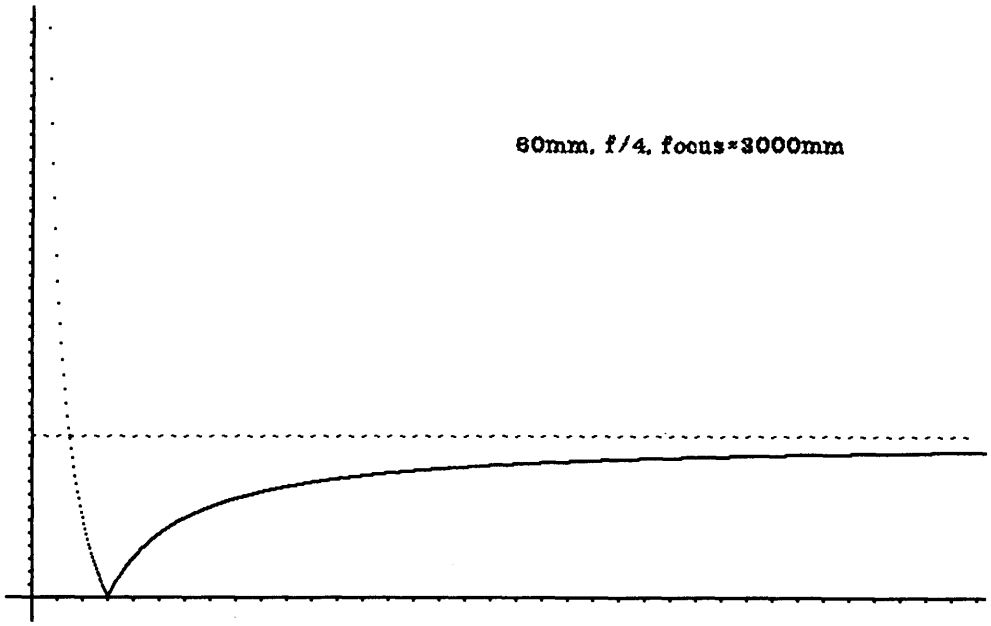

Figure 3-5: The geometrical circle-of-confusion function is very asymmetrical about the plane of best focus.

ambiguity exists if it is desired to recover distance by evaluating focus, as a given degree of defocusing could have resulted from the point in question being at one of two possible distances. Thus, depth-from-focus will require the camera to be focused closer than the closest object in a scene.[5]

[5]Or, in the case of very distant objects, it may be better to focus the lens at infinity. This issue will arise in a later section.

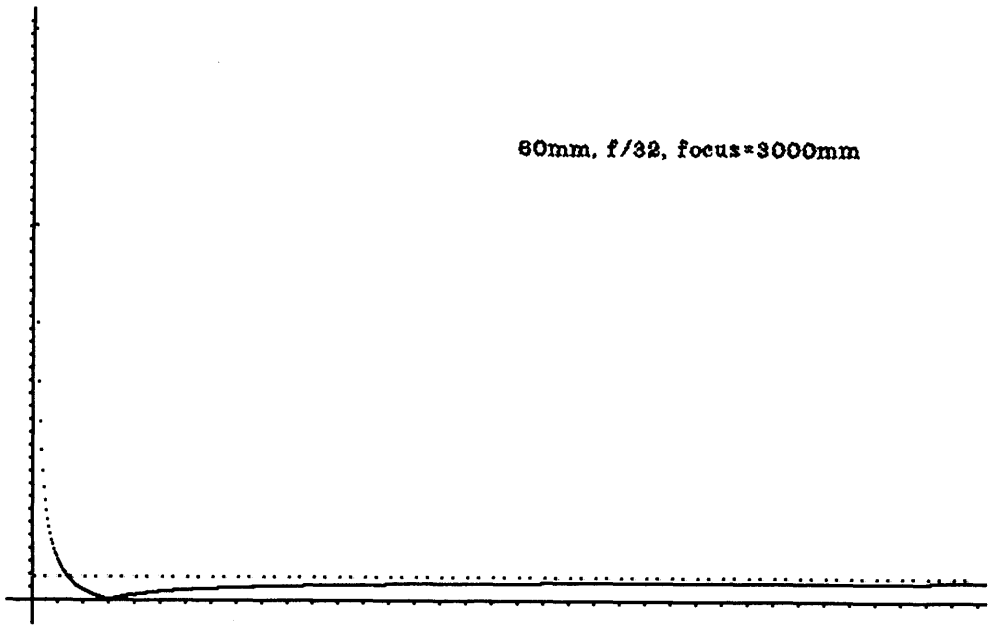

Figure 3-6: Closing the aperture lowers the asymptote of the function in the preceding figure.

3.4 Diffraction model

A more accurate description of the effects of defocusing requires taking diffraction into account. The scene must be modeled as a collection of point sources of light; for each it is necessary to solve the Huygens-Fresnel integral for the case of Fraunhofer diffraction of a point source of light by a circular aperture. The integral cannot be evaluated in closed form, but its value may be calculated as a series of functions U_n, first derived

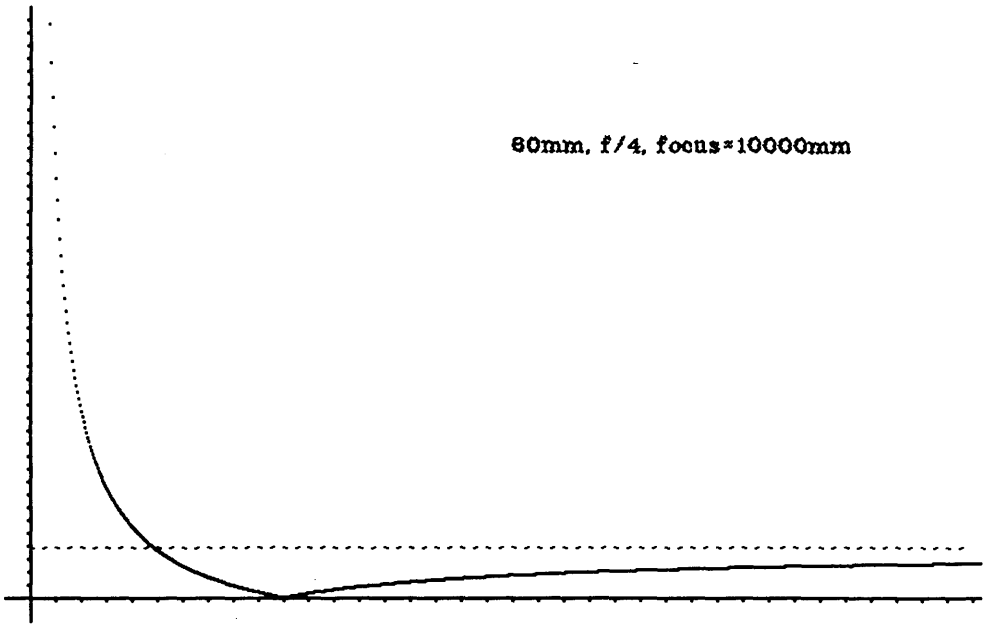

Figure 3-7: Asymptote of the circle-of-confusion function is also lowered by focusing at a greater distance.

by E. Lommel in 1885 [Born, 1970]:

$$U_n(u,v) = \sum_{s=0}^{\infty} (-1)^s \left(\frac{u}{v}\right)^{n+2s} J_{n+2s}(v), \tag{3.8}$$

where J_n are the Bessel functions of order n. The dimensionless variables u and v specify the position of a point within this distribution:

$$u = \frac{2\pi}{\lambda}\left(\frac{d}{2f}\right)^2 z, \tag{3.9}$$

$$v = \frac{2\pi}{\lambda}\left(\frac{d}{2f}\right) r = \frac{2\pi}{\lambda}\left(\frac{d}{2f}\right)\sqrt{x^2+y^2}. \tag{3.10}$$

Here r is radial distance on a plane away from the optic axis, while z is the distance by which the point source is out of focus. d and f are the aperture diameter and focal length, as before. The intensity distribution itself is

$$I(u,v) = \left(\frac{2}{u}\right)^2 \left[U_1^2(u,v) + U_2^2(u,v)\right] I_0. \tag{3.11}$$

The Lommel U_n functions, and thus this solution, converge quickly only for locations where $|u/v| \leq 1$, that is points closer to the focal plane than to the optic axis. For $|u/v| > 1$, it is computationally easier to use the equivalent expression in terms of the V_n functions:

$$V_n(u,v) = \sum_{s=0}^{\infty} (-1)^s \left(\frac{v}{u}\right)^{n+2s} J_{n+2s}(v), \tag{3.12}$$

$$\begin{aligned} I(u,v) \;=\; & \left(\frac{2}{u}\right)^2 \left[1 + V_0^2(u,v) + V_1^2(u,v)\right. \\ & \left. - 2V_0(u,v)\cos\left\{\frac{1}{2}\left(u+\frac{v^2}{u}\right)\right\} - 2V_1(u,v)\sin\left\{\frac{1}{2}\left(u+\frac{v^2}{u}\right)\right\}\right] I_0 \end{aligned} \tag{3.13}$$

This set of equations describes the distribution only for light of a particular wavelength λ, as illustrated in Figure 3-8. In white light, the maxima of one frequency may fall upon the minima of another frequency, with the result that the distribution becomes more uniform across v for a particular value of u (Figure 3-9). Further, real

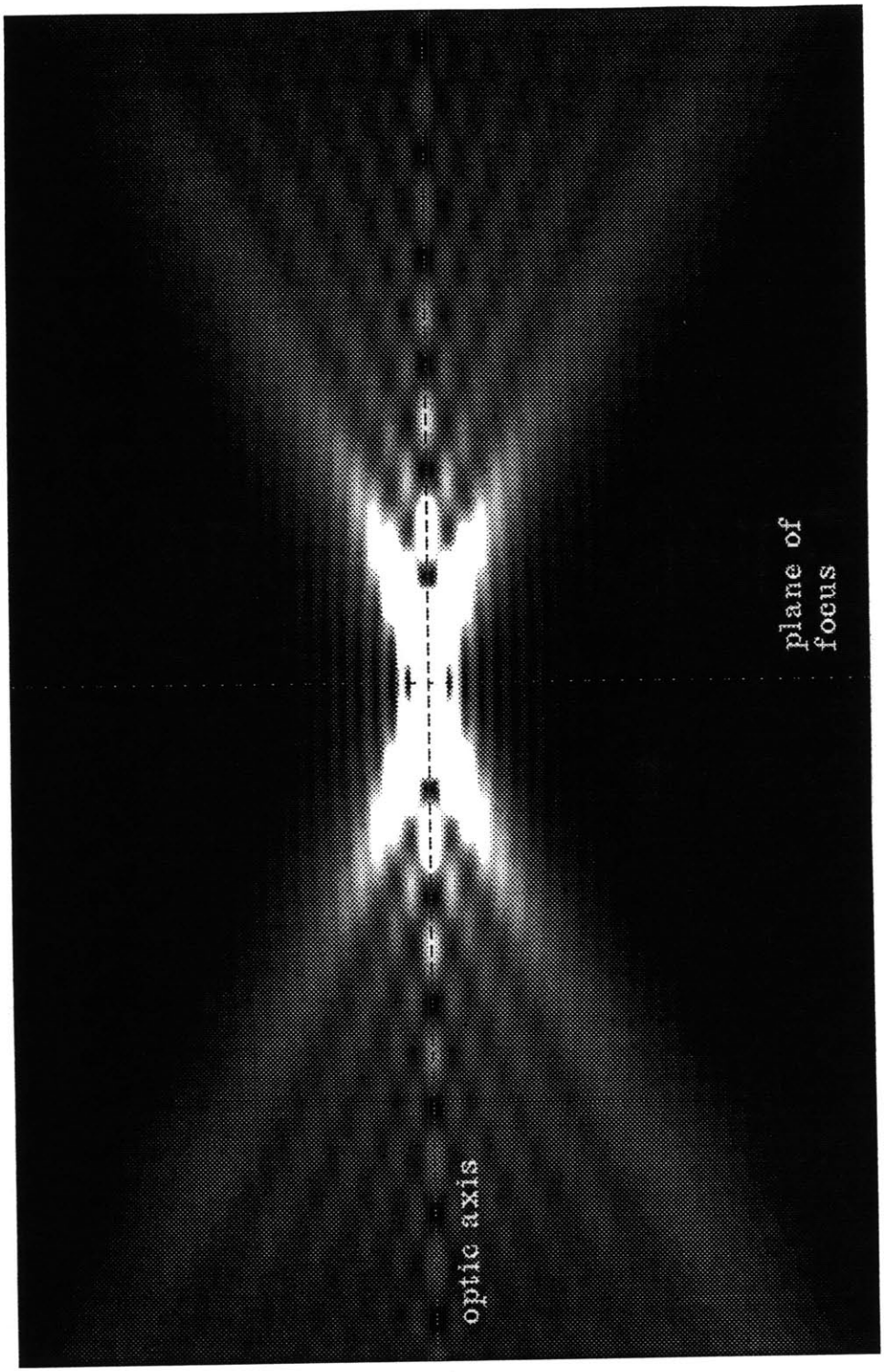

Figure 3-8: Intensity distribution around focus of a converging spherical wave diffracted through a circular aperture.

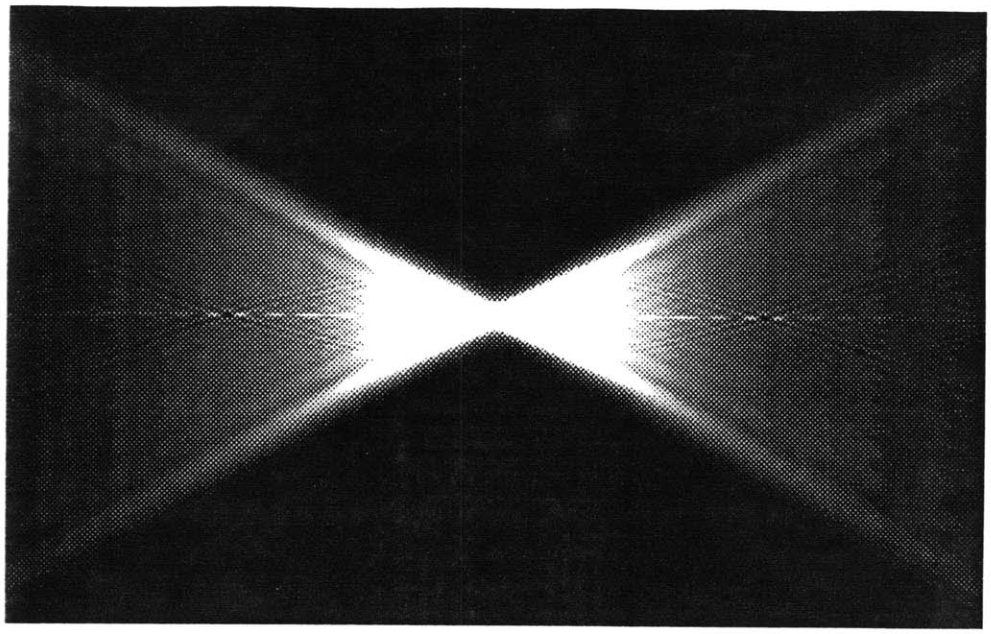

Figure 3-9: Intensity distribution of preceding figure integrated over wavelengths from 400 to 700nm. Image sensor characteristics will further reduce ripples in this distribution.

imaging systems typically have a resolution much coarser than the size of the ripples in this distribution,[6] and also tend to be light-sensitive through some significant volume (*e.g.* the emulsion coating on film or the thickness of a vidicon target). The result of these considerations is that the light distribution calculated above must be integrated across a range of wavelengths, as well as convolved with a three-dimensional point-spread function. A computer simulation of this process yielded a distribution which is largely uniform across some circular region and quickly rolls off to an almost-flat low

[6]For example, the maxima and minima in the v direction occur for v at multiples of π, or distances at multiples of λn. For the range of numerical aperture n commonly encountered in cameras (say $f/1.4$ to $f/32$) the ripples will be several orders of magnitude smaller than a pixel on a typical video camera.

level outside, very similar to the photographer's no-diffraction approximation. Except in the immediate vicinity of focus, the boundary of the central bright region is roughly identical with the edge of the "geometrical shadow" cast by the aperture, and thus for most real imaging situations the geometrical solution for the diameter of the circle of confusion should suffice.[7] A completely uniform circular distribution will be assumed throughout the remainder of the analysis.[8]

3.5 Evaluating degree of defocusing

The defocusing process may be defined as the convolution of image intensities $i(x, y)$ with a point-spread function $f(z)$ which is a function of distance to each point:

$$i_{defocused}(x, y) = i(x, y) * f(z). \tag{3.14}$$

This convolution with circles of increasing spatial extent corresponds in the frequency domain to multiplication by successively narrower low-pass filters [Herriott, 1958]. Horn was apparently the first to explore computer evaluation of focusing to infer distance

[7] Stokseth has done focusing analysis for monochromatic light and even in this situation concludes that, for defocus distances greater than 2λ, "...the shape of the two transfer functions is quite similar for low frequencies, and the geometrical [function] is a satisfactory fit to the exact [function] for frequencies inside the first positive sideband." [Stokseth, 1969](p. 1317) The difference occurs mostly at higher spatial frequencies (the "ripples" in the diffraction distribution which are smoothed out by the image sensor and the multiple frequencies of light).

[8] [Pentland, 1987b] chooses instead to use a Gaussian model of the light distribution. The sole change that this different assumption would make in the method to be described would be a modification to the series approximation used in the regression solution. This change has a minimal effect on the range values computed therefrom.

[Horn, 1968]; his idea was to use a discrete Fourier transform (DFT) to measure the degree of defocusing of a small patch of an image, and to use a servo-controlled lens to search for the optimal focus for that patch, at which point the high-frequency components of the power spectrum will be at a maximum. Lens magnification changes slightly with focus position, and processing software needs to take this effect into account when images from differing focus positions are compared. Jarvis has identified several easier-to-compute functions of pixel intensity which will maximize at best focus: entropy, variance, and sum-modulus-difference (the sum of the absolute values of differences among adjacent pixels) [Jarvis, 1983]. Similar measures, not DFT's, are used by automatic-focusing photographic and video cameras; since these measures are more easily confounded by image noise it is necessary to evaluate them over a very large image region for accuracy, and they are therefore not useful for deriving a large number of distance readings for a single image. Pentland explored using the high-frequency content of edges in a single picture of known focus distance for estimating the distance to the edges [Pentland, 1982], an approach which has more recently been expanded upon by Garibotto and Storace [Garibotto, 1987].

Another way of computing depth-from-focus is to vary the lens aperture rather than the focus distance. If a camera's nominal focus distance is at the front of a scene and a small lens aperture is used (say $f/22$) all points will be essentially in focus regardless of distance. If the lens is then opened up to a larger aperture (with

the addition of a neutral density filter to equalize the exposure[9]), comparative blur beyond the plane of best focus will be a monotonic function of distance from the camera. The short-depth-of-field image is modeled as the result of applying a distance-dependent blur function to the long-depth-of-field image, and corresponding regions in the two images are compared to measure the degree of blur and estimate the distance. Pentland proposed estimating the point-spread function by a linear regression technique [Pentland, 1987b] (though he actually implemented only a simplified algorithm which locally compared power in bandpass filtered versions of the images); a higher-order regression solution which gives greatly improved results will be developed, analyzed, and implemented below.[10]

3.6 Implementing a regression solution

Recall from above that

$$i_{defocused}(x, y) = i(x, y) * f(z). \tag{3.15}$$

The goal is to recover $f(z)$, and knowing its form, to calculate z. The recovery becomes practical if carried out in the frequency domain, where convolution transforms

[9]The density of this filter is calculated as follows. Since the density of optical filters is commonly expressed as the base-10 logarithm of the reciprocal of the transmittance, and since each opening of the aperture by one stop (dividing the f-number by $\sqrt{2}$, as from $f/32$ to $f/22$) doubles the light coming through the lens, the density of the filter must be 0.3 (or log(2)) per stop of compensation.

[10]See also [Bove, 1988]. Subbarao has performed an analysis of point-spread function variation in the event of incremental changes in focal length, focus distance, and aperture, but presents only limited experimental results through application of this method [Subbarao, 1988].

to multiplication. Therefore

$$F(\omega_x, \omega_y) = I_{defocused}(\omega_x, \omega_y)/I(\omega_x, \omega_y). \tag{3.16}$$

The preceding derivations show that $f(z)$ images a point source as a disk whose diameter increases as the point becomes increasingly defocused. Since this point-spread function is circularly symmetric, its transform should be circularly symmetric as well, and the problem may be solved in one dimension (that of radial distance) rather than two. The Fourier transform of a cylinder of height 1 and radius c is

$$F(\omega) = 2\pi c^2 \frac{J_1(\omega c)}{\omega c} \tag{3.17}$$

where J_1 is a Bessel function of the first order.[11] Thus, for a cylinder of radius c and height $1/\pi c^2$,

$$F(\omega) = 2\frac{J_1(\omega c)}{\omega c}. \tag{3.18}$$

Qualitatively, this function has a similar appearance to $\sin(x)/(x)$, consisting of a central peak with surrounding maxima and minima. J_1 may be expressed as a series expansion:

$$J_1(x) = \frac{x}{2} - \frac{x^3}{2^3 \cdot 1! \cdot 2!} + \frac{x^5}{2^5 \cdot 2! \cdot 3!} - \frac{x^7}{2^7 \cdot 3! \cdot 4!} + \frac{x^9}{2^9 \cdot 4! \cdot 5!} \cdots \tag{3.19}$$

[11] See [Hecht, 1974] for a method of deriving this transform.

so

$$2\frac{J_1(x)}{x} = 1 - \frac{x^2}{8} + \frac{x^4}{192} - \frac{x^6}{9216} + \frac{x^8}{737,280} \cdots . \tag{3.20}$$

Regression techniques will allow the estimation of blur circle diameter c by fitting a polynomial to the observed sample points of $F(\omega)$.[12] In order to make the calculations practical, the polynomial may be restricted to the first three terms of the series above. That is, for small x

$$2\frac{J_1(x)}{x} \approx 1 - \frac{x^2}{8} + \frac{x^4}{192}. \tag{3.21}$$

This may be rewritten in the form

$$(1 - y) = b_0 x^2 + b_1 x^4, \tag{3.22}$$

in which case, if y_i are the sample points observed, while Y_i are the "true" values of the desired function, free of error, the errors may be defined as

$$e_i = Y_i - y_i = Y_i - b_0 x_i^2 - b_1 x_i^4. \tag{3.23}$$

The sum of squares of the errors is

$$S = \sum (Y_i - b_0 x_i^2 - b_1 x_i^4)^2. \tag{3.24}$$

[12]See [Gerald, 1984] for a general discussion of polynomial regression.

In order to mimimize S with respect to b_0 it is necessary to solve $\partial S/\partial b_0 = 0$:

$$\frac{\partial S}{\partial b_0} = 0 = -2(\sum(1 - y_i) + b_0 \sum x_i^2 + b_1 \sum x_i^4). \tag{3.25}$$

The errors could likewise be minimized with respect to b_1 and the two resulting equations solved simultaneously, but since the desired form of the resulting polymomial is known, a substitution may be made: $b_1 = -b_0^2/24$. As a result,

$$\frac{b_0^2}{24} \sum x_i^4 - b_0 \sum x_i^2 + \sum(1 - y_i) = 0. \tag{3.26}$$

Solving this using the quadratic equation,

$$b_0 = \frac{\sum x_i^2 \pm \sqrt{(\sum x_i^2)^2 - \frac{\sum x_i^4}{6} \sum(1 - y_i)}}{\frac{\sum x_i^4}{12}}. \tag{3.27}$$

The smaller of these two roots proves to produce the correct approximation for $F(\omega)$. Recasting back to the original variables,

$$c = 2\sqrt{\frac{\sum \omega_i^2 - \sqrt{(\sum \omega_i^2)^2 - \frac{\sum \omega_i^4}{6} \sum(1 - F(\omega_i))}}{\frac{\sum \omega_i^4}{6}}}. \tag{3.28}$$

Of course, c will be here obtained in pixels (since the spatial frequencies are expressed in pixels/cycle), and a scale factor must be employed to convert the solution to a more useful measure of distance.

What must be done, then, is to take two-dimensional discrete Fourier transforms (DFT's) of corresponding small blocks of the short- and long-depth-of-field images, to collapse the two-dimensional spectra down to one dimension (that of radial frequency) by averaging all terms at given Cartesian distances from $(\omega_x, \omega_y) = (0,0)$, to divide the short-depth-of-field spectrum by the long-depth-of-field spectrum, and to use the regression equation to fit a curve to the resulting values (which represent the spectrum for the point spread function).[13] The parameters of the regression curve then give the distance to the image block according to the relation above (Figures 3-10, 3-11, 3-12, 3-13).

The DFT assumes that the input signal is periodic, a situation which does not hold for a patch of an arbitrary image. In effect, the transform that is obtained is that of an endless sheet of identical image blocks, arranged like postage stamps. As the left and right (and likewise top and bottom) edges of the block may not connect in a smooth manner, a discontinuity will appear which will introduce spurious high frequencies into the signal. The presence of this discontinuity may introduce undesirable artifacts into the range-estimation process.

One solution to the problem would be to mirror the image region both left-to-right and top-to-bottom. The resulting even symmetry would cause the DFT to contain

[13]It may have occurred to readers that the recovered spectrum is actually the absolute value of the function for which the regression solution was developed. As the fourth-degree polynomial approximation is a good fit only to roughly the point at which the curve first goes to zero, the regression fit is carried out only to the first minimum of the ratio. When experiments were done using higher-order curve fits (whose only-slightly-improved results failed to justify the greatly increased computation), the absolute value was taken into account in the process.

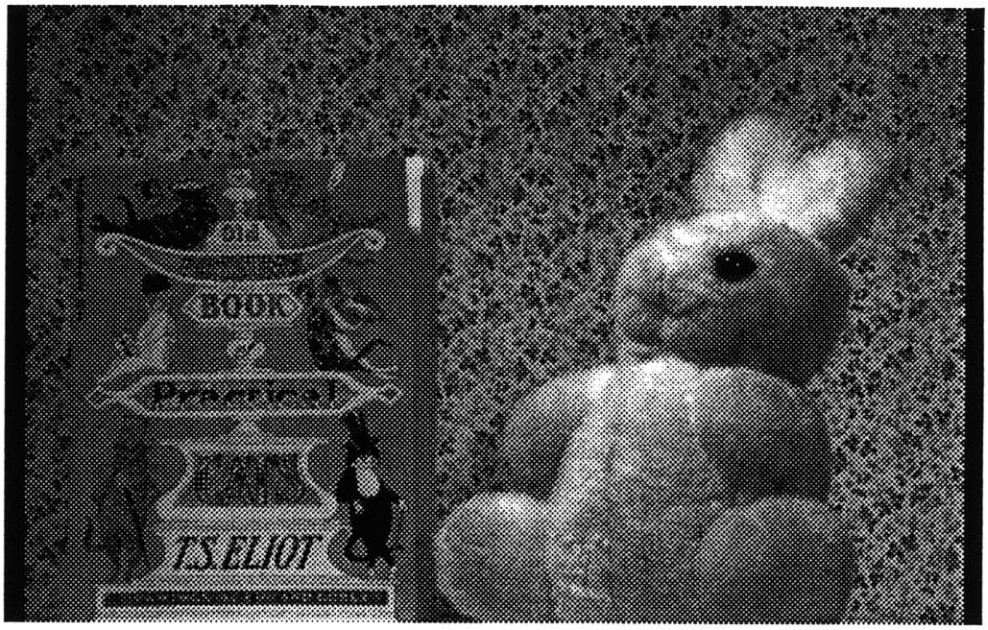

Figure 3-10: Long depth-of-field view of a scene.

only cosine terms, and the outcome would really be a discrete cosine transform, or DCT [Pratt, 1978]. Another method is the multiplication of the input block by a "window," a radially symmetrical function which peaks in the center of the block and goes to zero at the edges.

There is a space-frequency uncertainty associated with estimation of a space-varying spectrum [Gabor, 1946]. If the spatial resolution of a space-varying spectrum estimate from K samples of some signal $S(x, \omega)$ is denoted by Δx^o and the frequency resolution by $\Delta \omega^o$,

$$\Delta x^o = K, \tag{3.29}$$

Figure 3-11: Short depth-of-field view of same scene as Figure 3-10.

and

$$\Delta\omega^o \cong \frac{1}{K}. \tag{3.30}$$

Thus their product is constrained to be on the order of unity, the actual value depending upon the particular window employed, and the exact mathematical definition of Δx^o and $\Delta\omega^o$ used. It is common to define Δx^o and $\Delta\omega^o$ as the second central moments of the spatial and spectral windows (the latter of which is the transform of the former and is effectively convolved with the spectrum of the actual signal to yield the estimate). It can be shown that the minimum uncertainty product is obtained for a Gaussian window [Gardner, 1988], though numerous other windows are commonly

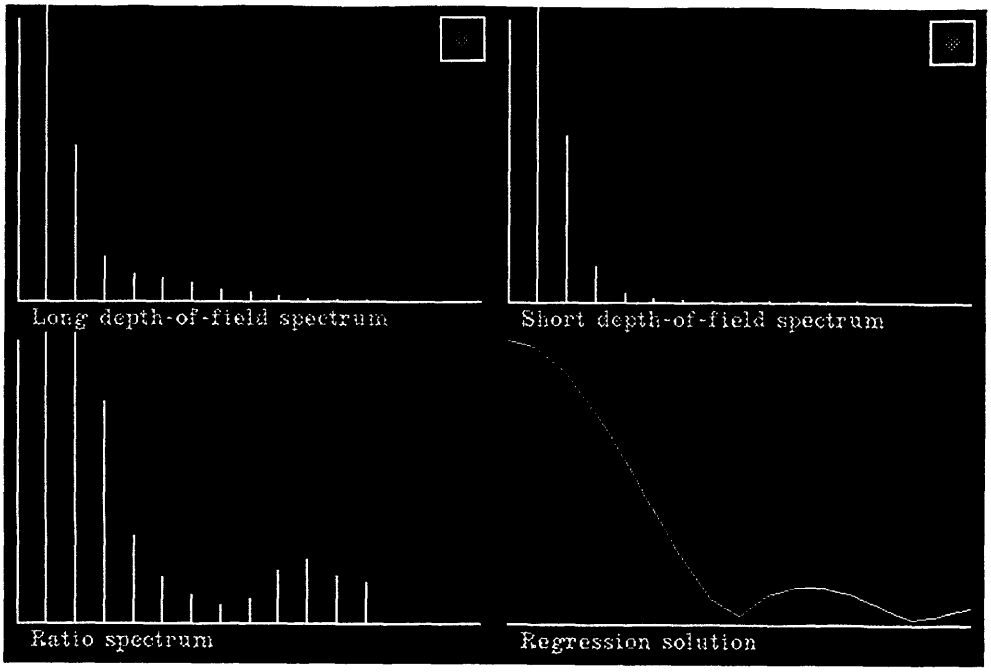

Figure 3-12: Illustration of the operation of the regression algorithm.

used either because they are easier to implement computationally or because they have various useful characteristics. A particular window shape will impart a statistical bias and variance to the spectrum estimate obtained, with a narrower spectral window reducing the bias at the expense of increased variance; making the best possible compromise depends upon knowing something about the statistics of the spectrum to be estimated. Full statistical analysis of optimal "window carpentry" is quite complex [Jenkins, 1968][Papoulis, 1977], but a Parzen window (which has significantly less variance – though somewhat more bias – than most other commonly used windows) has yielded good results for depth-from-focus purposes. This window is described as a

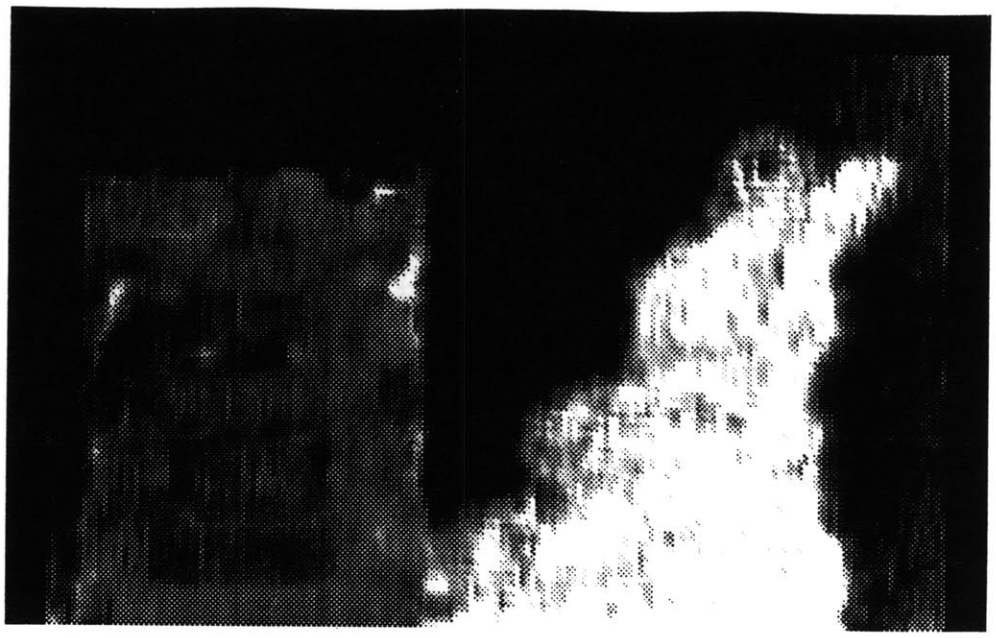

Figure 3-13: Range image computed from Figures 3-10, 3-11.

bell-shaped polynomial curve; if R is half the width of the image block to be windowed and r radial distance from the center:

$$w_{parzen}(r) = \begin{cases} 1 - 6\left(\frac{r}{R}\right)^2 + 6\left(\frac{r}{R}\right)^3, & r \leq \frac{R}{2} \\ 2\left(1 - \frac{r}{R}\right)^3, & \frac{R}{2} < r \leq R \\ 0, & r > R \end{cases} . \tag{3.31}$$

This uncertainty phenomenon suggests that depth-from-focus necessarily produces a range image of somewhat lower spatial (x, y) resolution than that of the images input to the process. Also, the spatial frequency components used to evaluate focus have a

wavelength of several pixels, leading to further uncertainty irrespective of the window size. These considerations suggest that it will be necessary to start with original images of higher resolution than the output range image desired. As an experiment, input film images taken with a 35mm still camera were scanned into the computer at a resolution of 1500-by-1500 pixels, and resulted in range images with significantly more resolution in x and y than those produced from the 525-line video images used in the rest of this work.

The size of the blocks transformed involves a tradeoff between obtaining enough regression terms to assure a good curve fit (and to estimate large blur circles), and localization of range detail. As the extent of the region being transformed decreases several changes result. The (x, y) resolution of the range image increases (since when using a large window, energy from an edge will affect patches near the edge, softening the z transition), but the z values become somewhat noisier (both because they are based on fewer pixels and contain greater uncertainty due to noise or misalignment and because the regression is being performed upon fewer spatial frequency components). The images illustrated in this thesis have been produced using either 32-by-32 or 64-by-64 pixel blocks. The computational requirements for the latter are fairly significant on a general-purpose computer, with most of the processing time spent performing the transforms, but hardware optimized for this sort of computation can process depth-from-focus images fairly quickly. A Sun TAAC-1 application accelerator computes a depth-from-focus image using 64-by-64 blocks in under five minutes, and a hardware

implementation of the algorithm would be faster still.

The regression solution developed above may also be employed with other bandpass image representations that are faster to calculate than DFT's, such as Laplacian or quadrature mirror filter pyramids [Burt, 1983][Adelson, 1987]. The power ratio in each level of a Laplacian-pyramid representation of the short- and long-depth-of-field images has been used to solve the regression equations, with useful results (Figures 3-14, 3-15). The major problem here is that there are fewer points available to the regression solution, and that they are less localized in frequency, leading to greater estimation error than for DFT's.[14]

3.7 Error analysis

There are a number of different sources of inaccuracy in the range images produced by the method of depth-from-focus outlined here. These may be divided into those due to the characteristics of the scene itself, those caused by the imaging system, and those resulting from the processing algorithm.

Image regions with no high-frequency content present a particular problem for all depth-from-focus techniques, in that little change occurs in these regions as a result of defocusing. If the lens is focused in front of the scene being imaged, these uniform areas

[14]Darrell and Wohn [Darrell, 1988] have also employed Laplacian pyramids for depth-from-focus, but they are simply searching among a group of images to find the one in which a particular image patch is best focused, in a manner similar to Horn's method.

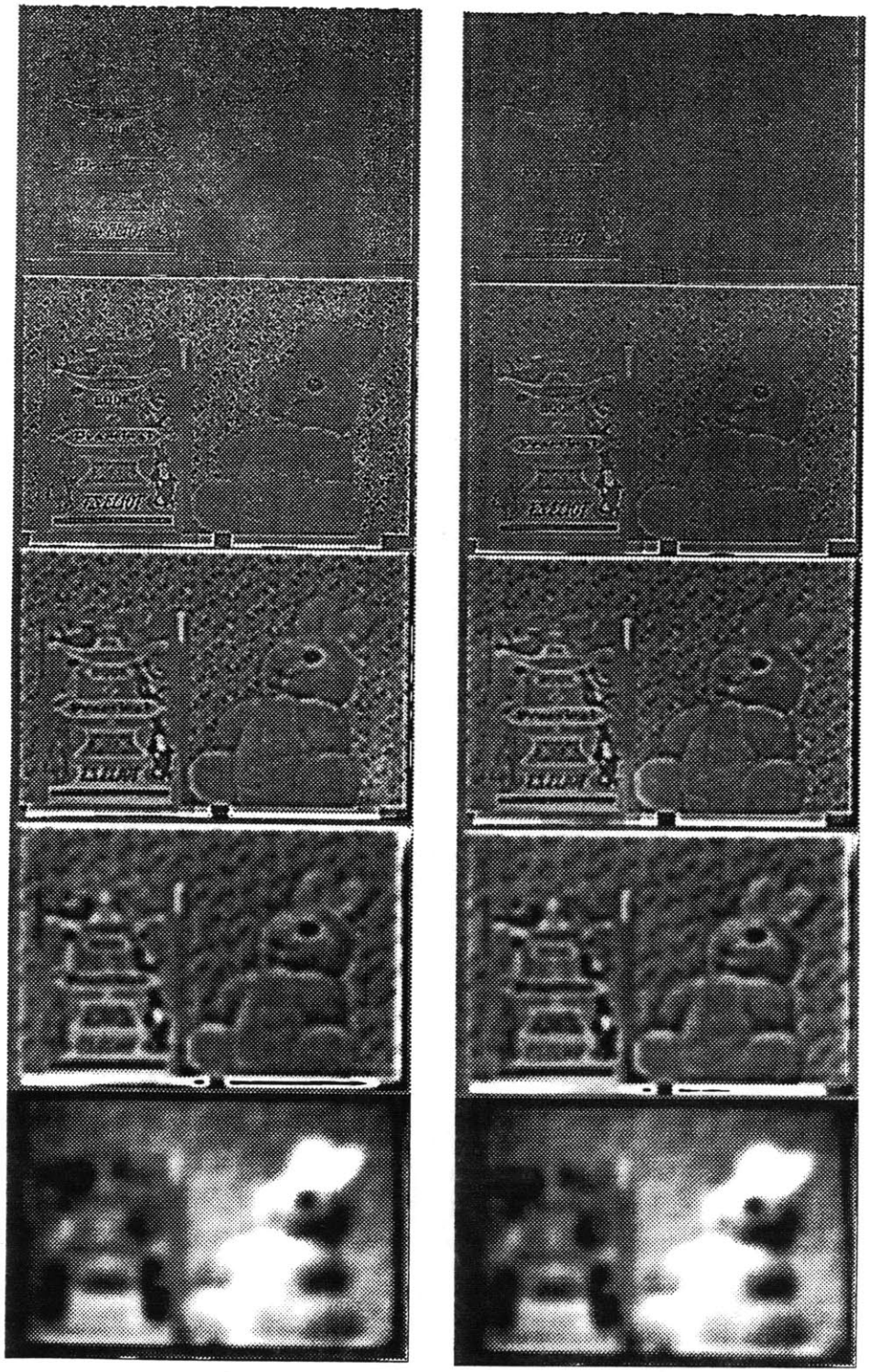

Figure 3-14: Laplacian pyramid representation of long- and short-depth-of-field views of a scene (magnified so that all levels are the same size for clarity).

Figure 3-15: Depth-from-focus computed by applying regression solution to pyramid representation of images.

will be interpreted as being closer to the camera than is correct. As the power spectrum of the long-depth-of-field image block is available to the software, these regions may be identified and labeled in advance. After the entire image has been processed, the holes may be filled in via the earlier-mentioned iterative methods originally developed for use on much sparser stereopsis images.

The imaging process introduces several significant sources of error as well. As the scene itself is not bandlimited in spatial frequency, the long-depth-of-field image may contain some degree of aliasing in highly-detailed regions. The defocused versions of these regions will contain no aliasing, and the results from dividing the power spectra

may not accurately reflect the point-spread function. Misalignment between the long-depth-of-field and short-depth-of-field images will also produce incorrect range values, especially if small image blocks are being transformed. The assumption that the long-depth-of-field image contains no blur whatever is not entirely true; points very far from focus may be defocused enough to affect the estimate of the blur circle. Noise in the image sensor is also a problem. At low spatial frequencies this is much less than image detail, but for most real images the image energy rapidly decreases with increasing frequency. At the same time, in a typical video camera the amount of noise power rises sharply at higher frequencies [Schreiber, 1986]. As the noise energy at various frequencies is more or less the same for the short-depth-of-field and long-depth-of-field images, examination of experimental results typically shows that that the ratio of the highest-frequency components in the two images is close to unity. That is, the recovered $F(\omega)$ looks like $2J_1(\omega c)/(\omega c)$ only up to the first few minima, above which it rises back to 1. This phenomenon implies either that curve-fitting to estimate the width of the point-spread function should be carried out only in the lower portion of the spectrum or that a noise model should be incorporated into the estimator (in any case, the fourth-degree polynomial approximation to the transform of the point-spread function holds only up to the first minimum). Noise is especially troublesome in dark regions with little detail, where it may overwhelm the legitimate image content. As the noise is unaffected by defocusing, dark areas may be assigned a range value too close to the focus distance of the lens. Like the uniform regions discussed in the preceding

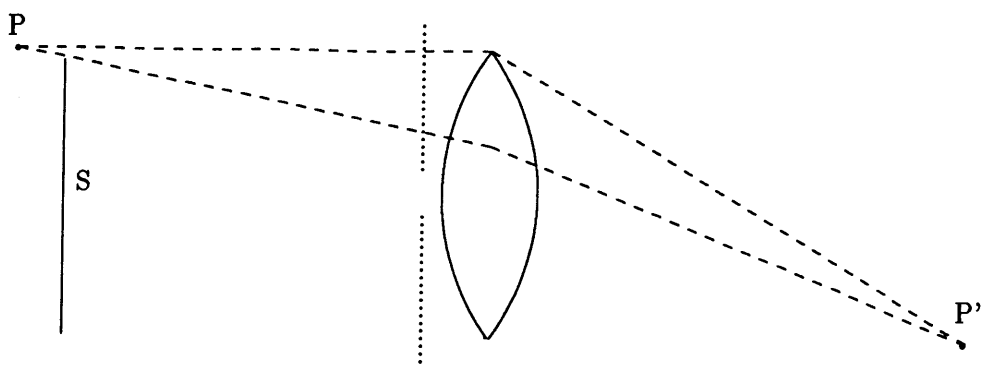

Figure 3-16: The "missing parts" problem for depth-from-focus. When the lens aperture (dotted lines) is fully open, point P images at P'. When the aperture closes down, surface S occludes P from the point of view of the lens. Typically the lens diameter will be many times smaller than the object distances, however, and this effect will be minimal.

paragraph, these locations may be identified and corrected through relaxation.

A small "missing parts" phenomenon, similar in principle to that in stereopsis, is illustrated in Figure 3-16. When the lens aperture is fully open, point P focuses at P'. When, however, the iris (dotted lines) is closed down surface S occludes P from the point of view of the lens. Thus the points imaged vary to some degree with the aperture, providing a further theoretical limit to this sort of depth-from-focus process. That this is a minimal problem at object distances much greater than lens diameter is easily seen from the diagram, however.

Additional inaccuracy is computational. The effects on resolution of using a fairly large window have already been discussed. The lens-and-aperture model employed

may not be strictly correct for a given lens, and the polynomial approximation to the transform of the point-spread function adds yet more inaccuracy to the regression process.

By making a few assumptions it is possible to approach the accuracy problem in an analytical manner, leading to a better understanding of the effects of inaccuracy in the blur-circle estimation process. The results can then be compared with actual error measured under experimental conditions.

The calculus of errors says that if a function f of n variables $f(x_1, \ldots, x_n)$ is calculated with the approximate values $(a_1, \ldots, a_n)$, where each $a_i = x_i + \epsilon_i$, then the exact error ϵ_f of f (assuming the error terms are statistically independent) is given as

$$\epsilon_f = \frac{\partial f}{\partial x_1}\epsilon_1 + \cdots + \frac{\partial f}{\partial x_n}\epsilon_n. \tag{3.32}$$

Gauss' law of propagation of errors further states that if the standard deviations of the individual error terms are known then the standard deviation of the function is

$$\sigma_f = \sqrt{\left(\frac{\partial f}{\partial x_1}\right)^2 \sigma_1^2 + \cdots + \left(\frac{\partial f}{\partial x_n}\right)^2 \sigma_n^2} \tag{3.33}$$

[Gellert, 1975].

Rearranging the geometrical circle-of-confusion expression to give the recovered

range U_y in terms of the diameter c, in the region past the focus distance V_x

$$U_y = \frac{fV_x}{V_x - f - nc}. \tag{3.34}$$

Since f, V_x, and n are constants, all the error in this expression comes from the error in c. Replacing c by $(c + \epsilon_c)$,

$$\epsilon_{U_y} = \frac{\partial U_y}{\partial c}\epsilon_c, \tag{3.35}$$

so

$$\epsilon_{U_y} = \frac{nfV_x}{(V_x - f - n(c + \epsilon_c))^2}\epsilon_c = \frac{n}{fV_x}U_y^2\epsilon_c. \tag{3.36}$$

By a similar process

$$\sigma_{U_y} = \frac{n}{fV_x}U_y^2\sigma_c. \tag{3.37}$$

That is, as distance from the camera increases, the error in U_y increases as the square of the distance. The error plot as a function of distance for a constant error in blur circle estimation is shown in Figure 3-17.[15]

Given a model for computing error in z from error in c, the next step is to relate error in c to image content. A useful bound on the variance of an estimate of this sort is the Cramér-Rao inequality [Van Trees, 1968]. Given additive noise, the observed short

[15] In [Pentland, 1987b] an analysis is done for the case of constant percentage error, as suggested by the results of experiments on human vision. This assumption leads to the unlikely result that as $U_y \rightarrow U_x$ the error in U_y due to the error in the diameter estimate goes to zero, a conclusion which is not necessarily appropriate for analyzing a regression estimator.

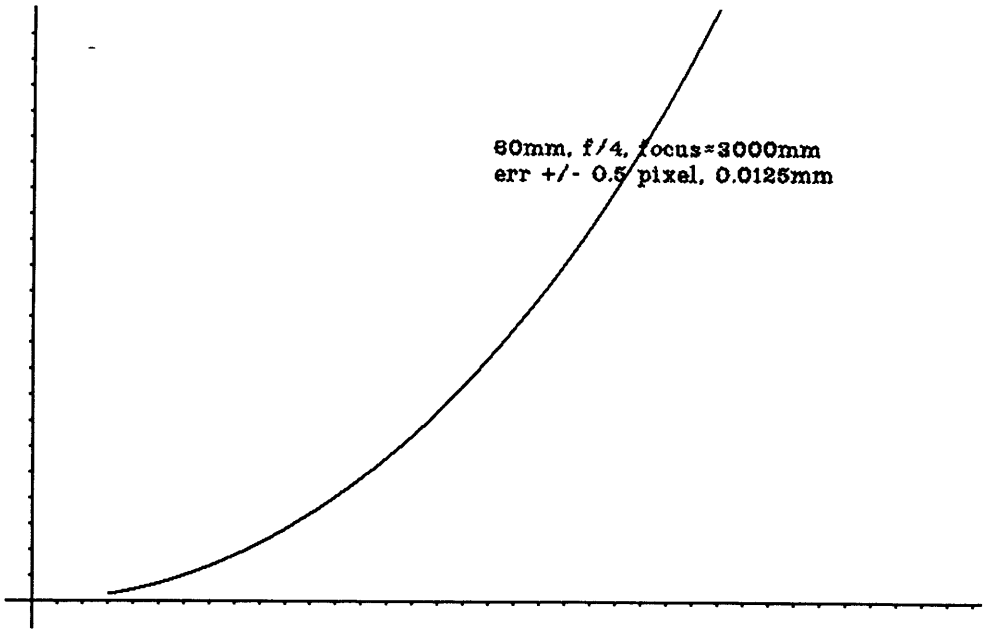

Figure 3-17: Predicted error in z for a constant error in estimating c. Both scales are in meters.

depth-of-field spectrum $r(\omega)$ will be modeled as the sum of some $s(\omega, c)$ and white, zero-mean Gaussian noise $n(\omega)$ with standard deviation σ_n:[16]

$$r(\omega) = s(\omega, c) + n(\omega), \quad 0 \leq \omega < K. \tag{3.38}$$

The Cramér-Rao lower bound on minimum mean-square error of an estimate of the

[16] As noted above, the assumption of white, Gaussian noise is probably not strictly correct. If the actual spectral distribution of noise were known, it would be here regarded as the product of white noise and some transfer function $H(\omega)$. Whiteness is also technically violated by the collapse of two-dimensional spectra to radial frequency, as not all radial frequency terms result from the same number of terms in the original spectrum, and thus variance is a function of ω.

random variable c is

$$var[\hat{c} - c] \geq \left[-E\left(\frac{\partial^2 \ln(p_{\mathbf{r},c}(\mathbf{r}_0, c_0))}{\partial c^2} \right) \right]^{-1} \tag{3.39}$$

where $\mathbf{r}$ is the vector of observations of r, in this case the observations of the ratios of spectral terms, and $\hat{c}$ is the estimate of c. The expression for $s(\omega, c)$ may be split:

$$s(\omega, c) = s(\omega)F(\omega, c). \tag{3.40}$$

Since $\mathbf{r}$ and c are independent, their joint probability density function may be rewritten:

$$p_{\mathbf{r},c}(\mathbf{r}_0, c_0) = p_{\mathbf{r}}(\mathbf{r}_0)p_c(c_0). \tag{3.41}$$

Proceeding further requires some knowledge of the scene and camera parameters. It might be assumed that all c are equiprobable between some minimum and some maximum value determined by the imaging situation.[17] Thus the probability density function for c is

$$p_c(c_0) = \begin{cases} \frac{1}{c_{max} - c_{min}}, & c_{min} \leq c \leq c_{max} \\ 0, & otherwise \end{cases} . \tag{3.42}$$

[17] More correctly, one might assume that all z are equiprobable, and compute the PDF for c based upon this knowledge and the shape of the $c(z)$ curve for the particular lens parameters employed. For a scene in which the range of c is fairly limited, assuming a uniform PDF for c is not too unreasonable given a uniform PDF for z.

The probability density function for the noise component n is

$$p_n(n_0) = \frac{1}{\sqrt{2\pi}\sigma_n} \exp\left(-\frac{n_0^2}{2\sigma_n^2}\right). \tag{3.43}$$

If $s(\omega)$ is assumed to be known, then the PDF for the observation vector may be expressed:

$$p_{\mathbf{r}}(\mathbf{r}_0) = \left(\frac{1}{\sqrt{2\pi}\sigma_n}\right)^K \exp\left(-\frac{1}{2\sigma_n^2}\sum_{i=1}^{K}(r(\omega_i) - F(\omega_i, c)s(\omega_i))^2\right). \tag{3.44}$$

After multiple substitutions and simplifications, the Cramér-Rao bound becomes

$$var[\hat{c} - c] \geq \left[\frac{1}{\sigma_n^2}\sum_{i=1}^{K}\left(s(\omega_i)\frac{\partial F(\omega_i, c)}{\partial c}\right)^2\right]^{-1}, \tag{3.45}$$

an expression which allows computing the lower bound for the variance in the estimate of c given a particular $s(\omega)$, a measure of noise content, and a presumed-correct model for the blur function $F(\omega, c)$. That this bound is very optimistic and will not likely be approached by an actual implementation is due to several causes, in particular: $F(\omega, c)$ is not a perfect model, the noise model assumed is not a perfect model, $s(\omega)$ is not actually known – instead there is only a long-depth-of-field spectrum which itself contains some amount of blur, noise, and aliasing,[18] and (as observed earlier) the

[18] Aliasing content can be included in the expression for $r(\omega)$ as an additional term $s(W - \omega)$ where W is the sampling frequency.

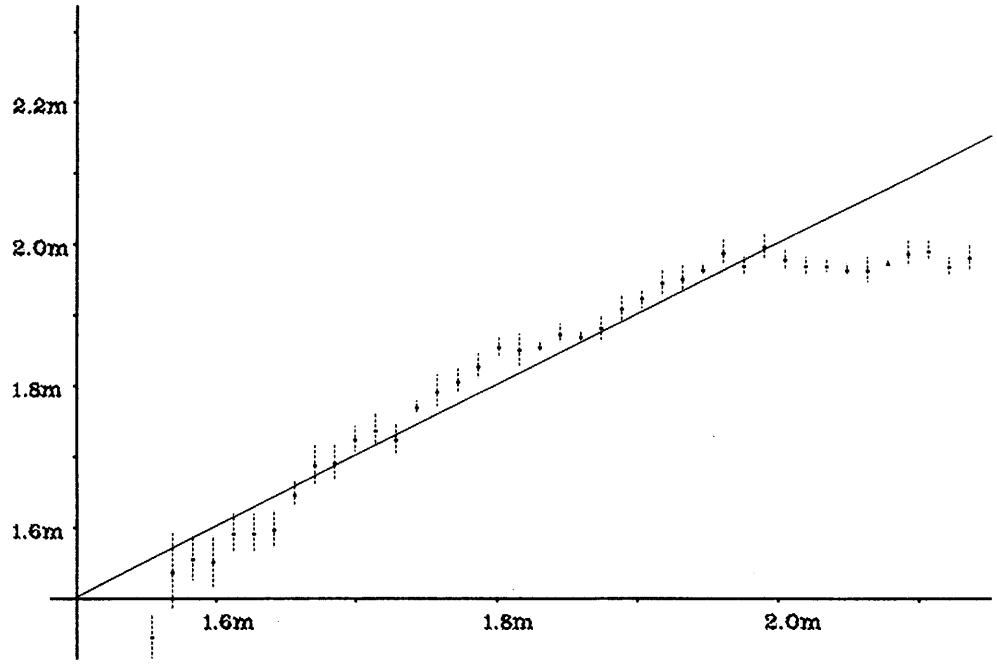

Figure 3-18: Results of an experiment in which mean and standard deviation were computed across equidistant regions of a depth-from-focus range image for a sloping surface. Horizontal axis is actual distance, vertical is estimated. Solid line represents the position of the test surface. 60mm lens, apertures $f/5.6$ and $f/22$, focus 1.3m.

spectrum estimates result from a windowing process that introduces bias, variance, and spatial uncertainty.

In Figure 3-18, an experiment has been done where depth-from-focus was computed for a sloping, patterned plane, and mean and standard deviation were calculated across equal regions of the range image known to be at the same distance from the camera. In this plot, the standard deviation clearly increases for points in the foreground, a phenomenon that might be explained by the fact that the differences between the long- and short-depth-of-field spectra for small degrees of defocusing occur in the high spatial

frequencies, where the signal-to-noise ratio is typically lowest. Some other observations about this graph are in order: in the extreme foreground, where the blur is small, the software seems to err on the low side, while in the distance the blur circle is approaching its asymptote and again the regression software has trouble determining the correct range. This is also the region in which the long-depth-of-field image starts becoming perceptibly blurred, and the estimated blur circle may be a little too small here (a problem which could have been reduced by using a smaller aperture). In the central region, however, the accuracy is surprisingly good; it appears that at least three bits of range resolution are available within this area (of course, by proper adjustment of focus and apertures the location and extent of this area may be varied). The nonlinearity in the central region is possibly the product of an incorrect match of the software defocusing model and the actual point-spread function of the lens employed, and could be corrected quickly via a look-up table or more properly by careful measurements on the lens and construction of a more appropriate regression polynomial.

The preceding discussion suggests that there is some range of blur circle diameters that this software is capable of estimating accurately for a given transform size, window, regression model, *et cetera.* Recalling the circle-of-confusion function derived at the beginning of this chapter, it should be noted that the asymptotic shape of the far-field part of the function and the fact that the asymptote drops as a lens' focus moves farther away make the far-field part of this curve less than ideal for estimating range accurately at large distances. In applications where it is desirable to distinguish objects at, say, 50

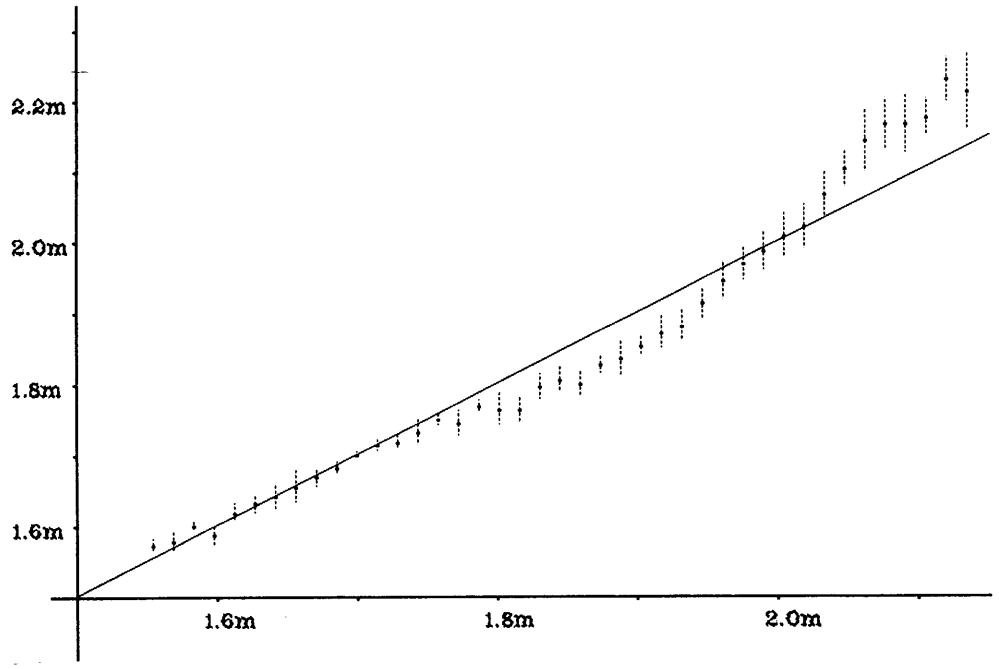

Figure 3-19: Experiment of preceding figure repeated for near-field part of blur function. Same parameters, except focus is 2.5m.

meters from those at 75 meters employing the near-field part of this curve will prove to be more productive (recall Figures 3-5, 3-6, 3-7).

The experiment of Figure 3-18 has been repeated using the near-field part of the circle-of-confusion curve, focusing the lens just beyond the farthest point visible in the scene. As the regression software apparently tends toward estimating small blur circles as too small, it may be predicted that in this case points near the far end of the plane will be estimated as being too far away. This prediction is borne out by the observed data (Figure 3-19). Again, the standard deviation of the data seems to increase as points approach focus.

3.8 Camera hardware

Gathering the range information for a moving scene requires taking the long- and short-depth-of-field images at the same time. One simple solution is to have two cameras, identical except in iris openings and neutral-density filters, looking at a scene through a half-silvered mirror. In practice, problems are encountered with assuring that the focus of both lenses always remains the same. An easier-to-use arrangement might involve only one front lens element with a beamsplitter behind it, imaging upon two different sensors through different apertures.

In order to create a very compact camera apparatus whose output could be recorded on a single videotape recorder or film frame, a lens assembly was built in which the long- and short-depth-of-field views are simultaneously imaged side-by-side (Figures 3-20, 3-21). As the diagrams show, there are actually two front elements with a small horizontal offset, which results in a slight stereopsis effect. The offset is so small (9mm), however, that for scenes at reasonable distances from the camera there is no significant parallax disparity between the views. This unit focuses by sliding rather than rotating, so the two images remain aligned in the same way regardless of the focus position. The assembly results in images with a somewhat unusual aspect ratio (2 units wide by 3 high), though an anamorphic 2:1 squeeze could be incorporated into the optics (and undone in the processing software) to correct the aspect ratio objection. Also, as this unit employs only a cemented doublet on each side its images are not as sharp

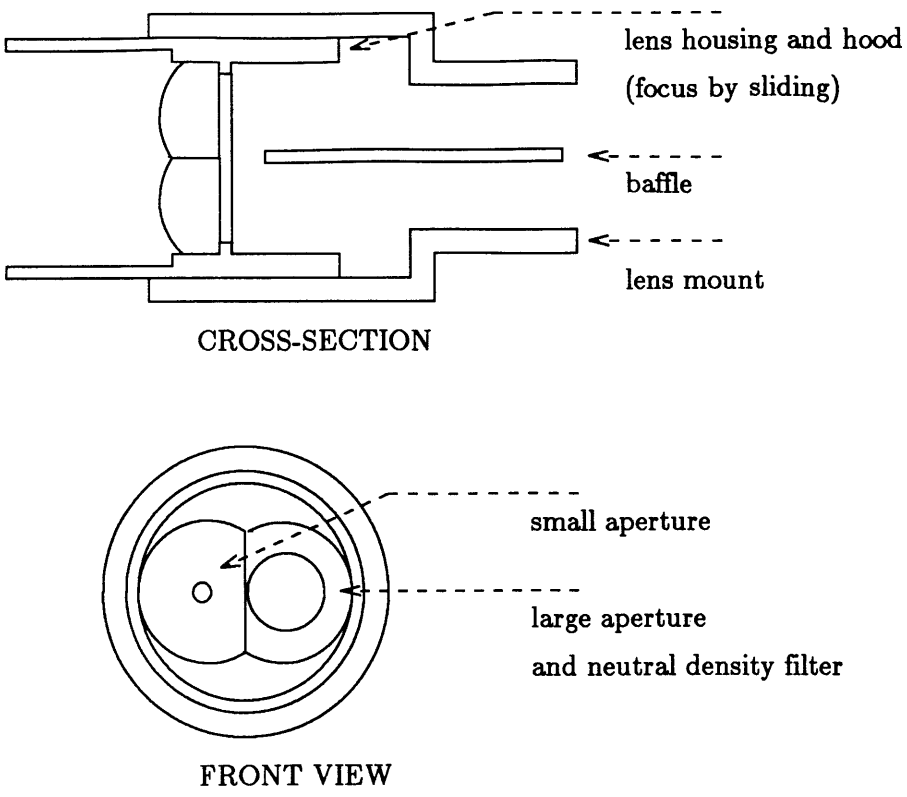

Figure 3-20: Two-element lens images both views of a scene simultaneously.

(nor the depth-from-focus results quite as accurate) as those resulting from the Nikon 35-millimeter-camera lenses employed earlier in this chapter. Still, this lens assembly is quite rugged and compact, and provides depth-from-focus data more than adequate for many purposes.

If an object is moving rapidly, the motion blur at the camera may impair the evaluation of defocusing. In such cases, a camera with a high-speed electronic or mechanical shutter will improve the accuracy. This issue will assume more importance in the discussion of motion estimation.

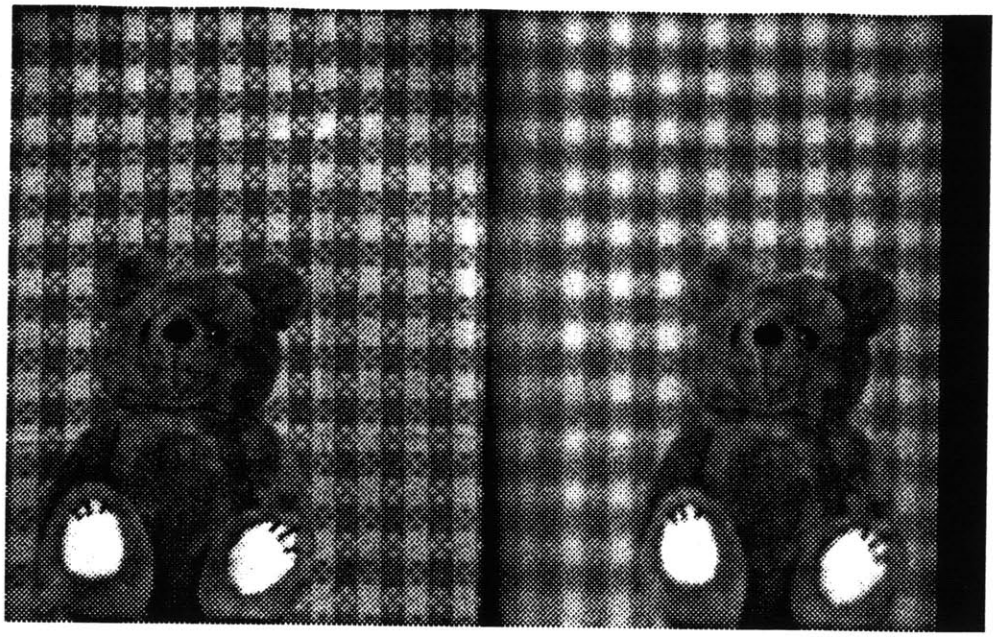

Figure 3-21: Image from two-element lens.

3.9 Chapter summary

The principles behind the phenomenon of focusing have been explained, and a method of estimating range by evaluating degree of blur has been developed and analyzed. A lens assembly has been designed allowing range data to be gathered for moving subjects in real time using an ordinary film or video camera.

Chapter 4

Motion and structure

In the midst of chaos there was shape; this eternal passing and flowing (she looked at the clouds going and the leaves shaking) was struck into stability.

Virginia Woolf

To the Lighthouse

4.1 Approaches

If two views are taken of a scene with a displacement in time, motion (either of the camera or of objects in the scene) permits the recovery of range. Helmholtz over a century ago recognized that the different velocities of two-dimensional projections of moving points carry depth information [Helmholtz, 1924]. Gibson and Gibson later coined the term "optical flow" to describe the image projection of three-dimensional motion vectors [EGibson, 1959]. While optical flow fields in general contain information about both structure and kinematics of scenes [Prazdny, 1983], a much easier-to-solve problem

is recovering structure when motion is known in advance. Depth-from-camera-motion involves taking a series of views of a still scene at small, known increments of camera displacement, and has been extensively studied [Moravec, 1979][Rives, 1986][Matthies, 1987]. The advantage of this technique over stereopsis is that the small displacements employed simplify the finding of correspondence between views. Commonly the camera motion is strictly along only one image axis, further simplifying the computation, as correspondence searching can be done in only one dimension.

Techniques for calculating the motion of an image point across multiple frames of a motion sequence fall into three classes: region-matching methods, transform-domain methods, and spatiotemporal constraint methods. Region-matching methods, which are computationally similar to those used in stereopsis, likewise seek correspondence of either image patches [Ninomiya, 1982] or features such as edges at various scales [Hildreth, 1984] between frames. Sub-pixel accuracy in correlation-based methods is sometimes created by interpolating additional points within a sample block.

A spatial displacement of a region between frames will appear as a phase shift of spatial frequencies if a Fourier transform is applied to the images. Lo and Parikh [Lo, 1973] and Haskell [Haskell, 1974] did some early work with detecting these phase shifts to evaluate the corresponding motion. An improved method is phase correlation [Kuglin, 1975][Pearson, 1977]. The phase correlation function is computed by performing a discrete Fourier transform of each image, forming the cross-power spectrum by multiplying the real part of each component of the first by the imaginary part of the

corresponding component in the second, and then taking the inverse transform of the result. Ideally, the resulting array of numbers (the "correlation surface") will have a strong peak at the location corresponding to the displacement between the two images. The sharpness of this peak can be improved by normalizing the amplitudes of all the frequency components before computing the inverse transform, and filtering and interpolation can be performed upon the phase correlation surface to increase the accuracy with which the center of the peak can be located. Frequency-domain-based methods seem to have found their greatest use in very accurate registration of entire images which differ only in global displacements, but Girod has obtained promising motion-estimation results using a similar operation locally [Girod, 1989].

Spatiotemporal constraint methods assume that the intensity of an object point does not change with time[1] and solve a set of linear equations in the spatial and temporal gradients of image intensity in order to determine optical flow. Essentially, if image intensity is I,

$$\frac{\partial I}{\partial x}\frac{dx}{dt} + \frac{\partial I}{\partial y}\frac{dy}{dt} + \frac{\partial I}{\partial t} = 0. \tag{4.1}$$

These techniques can be quite computationally efficient, but require very good estimates of the image gradients, and may require application at multiple spatial scales in order to achieve accurate estimates for large motions [Martinez, 1986].

The converse of depth-from-camera-motion is estimating three-dimensional object

[1]Or, equivalently, that it changes slowly relative to the change in local image intensities due to object motion. See [Gennert, 1986b] for a discussion of relaxing this requirement.

motion for a stationary camera and moving scene. In the case where the scene consists of only one object's surface, of course, the mathematics is the same. But even for a scene containing a single moving object in a stationary environment, some segmentation of elements is needed so that their motions may be calculated separately. If an optical flow field is to be used to recover the motions of objects, an additional assumption is usually added to simplify calculation. A rigid object's motion may be uniquely specified in terms of translational and rotational components [Halfman, 1962], and if the scene may be segmented into pieces for which rigid-body motion approximately holds, then the translation and rotation may be recovered for each rigid piece [Horn, 1986]. Information from another source (such as depth-from-focus data) might be used for this segmentation, or the optical flow fields may themselves be partitioned with some success if noise is small [Adiv, 1985]. Indeed, if a parametric description of a rigid object's surface shape is available, its motion may be estimated directly from the gradients and surface normal vectors without intermediate optical flow calculations [Negahdaripour, 1987]. An idea which does not appear to have been investigated in detail by other researchers is the use of range data from a process like depth-from-focus along with spatiotemporal gradients to estimate rigid-body motion of scene elements, and in turn the application of structure-from-motion to refine the initial range estimates. Such a process will be developed in the following sections.

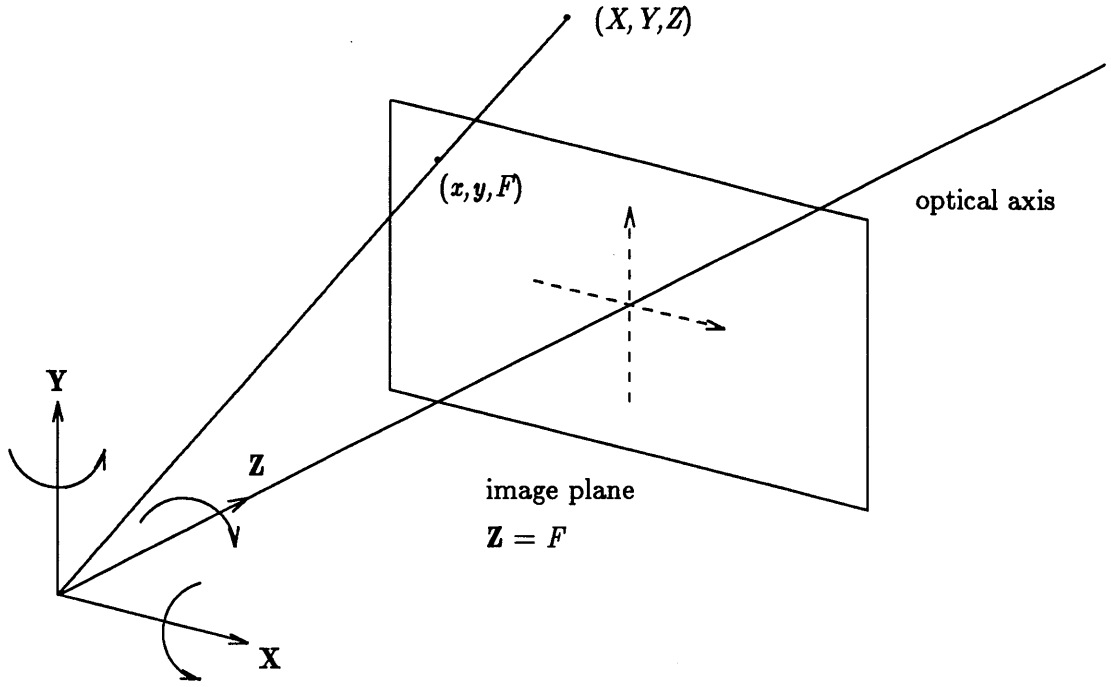

Figure 4-1: The pinhole camera model. A point at (X, Y, Z) is projected onto the image plane at (x, y, F).

4.2 Camera model and optical flow

A camera model will be necessary in order to develop motion equations for real images. If the distance to the scene is significantly greater than the focal length of the lens and lens distortions are not significant, a pinhole camera model [Sobel, 1974] will simplify derivations (Figure 4-1). Let the origin of a Cartesian coordinate system be at the center of projection of the viewing pyramid, with the optical axis collinear with the z axis. Several different units of measurement are available: either absolute distance such as millimeters, or camera-relative measures such as focal lengths of the lens or pixels on

the camera's image sensor. If the latter is assumed, then an image of the scene will be formed on the plane $z = F$, where F is the focal length of the lens expressed in sensor pixels (this number is equal to the focal length in millimeters multiplied by the ratio of the number of pixels across the sensor to the width in millimeters of the sensor's active area). A scene point $\mathbf{R} = (X, Y, Z)$ is then imaged on the sensor at $\mathbf{r} = (x, y, F)$, where x and y are image coordinates, and are calculated from the scene coordinates as follows:

$$\mathbf{r} = \frac{F}{\mathbf{R} \cdot \hat{\mathbf{z}}} \mathbf{R}. \tag{4.2}$$

When a point $\mathbf{R}$ moves in space, the camera sees a two-dimensional projection of its motion, known as its optical flow. The velocity of the point $\mathbf{r}$ is

$$\frac{d\mathbf{r}}{dt} = \frac{d}{dt} \frac{F}{\mathbf{R} \cdot \hat{\mathbf{z}}} \mathbf{R}. \tag{4.3}$$

This derivative

$$\frac{d\mathbf{r}}{dt} = \frac{F}{\mathbf{R} \cdot \hat{\mathbf{z}}} \frac{d\mathbf{R}}{dt} - \frac{F}{(\mathbf{R} \cdot \hat{\mathbf{z}})^2} \left(\frac{d\mathbf{R}}{dt} \cdot \hat{\mathbf{z}} \right) \mathbf{R} \tag{4.4}$$

may be rewritten with the substitution $\mathbf{r} = (F/\mathbf{R} \cdot \hat{\mathbf{z}})\mathbf{R}$ and rearranged:[2]

$$\frac{d\mathbf{r}}{dt} = \frac{1}{\mathbf{R} \cdot \hat{\mathbf{z}}} \left(\hat{\mathbf{z}} \times \left(\frac{d\mathbf{R}}{dt} \times \mathbf{r} \right) \right). \tag{4.5}$$

[2]The cross-product notation here and in the following two sections follows that of Negahdaripour and Horn [Negahdaripour, 1987], with the exception of a non-unity focal length.

The disappearance of the leading F factor seems rather a curious result, in that this expression appears to imply that the velocity of the projected image point is independent of the focal length. An examination of the elements of $\mathbf{r}$ (which themselves depend upon F), however, clears up the apparent paradox.

4.3 The rigid body assumption

In order to reduce the number of parameters describing the motions of points in a scene, many researchers suggest collecting points into groups which will move together in a more-or-less rigid manner [Webb, 1982][Ballard, 1983][Netravali, 1985][Broida, 1986]. Under this assumption, non-rigid objects, such as a human arm, are considered as jointed sets of rigid pieces.[3]

By Chasles' theorem [Whittaker, 1937], the most general displacement of a rigid body is a translation plus a rotation. The translation may of course be expressed as a three-dimensional vector, but a finite rotation cannot properly be represented as a vector, as the result of two rotations is the product of two matrices, and matrix multiplication is not commutative. However, if infinitesimal rotations (such as might occur between two frames taken by a video camera[4]) are considered, the change in

[3]Other possible assumptions include what Ullman calls *incremental* rigidity [Ullman, 1983], and totally-deformable objects. These will be discussed later as potential extensions to this process. What is important is to have *some* assumption that reduces the number of unknowns, overconstraining the solution for robustness in the presence of noise. Later, when points seen at different times are assembled into object models, the rigid-body assumption will enable calculating the motion of points not seen by the camera based upon that of visible points.

[4]Here it is assumed that the motion is slow compared with the camera's frame rate.

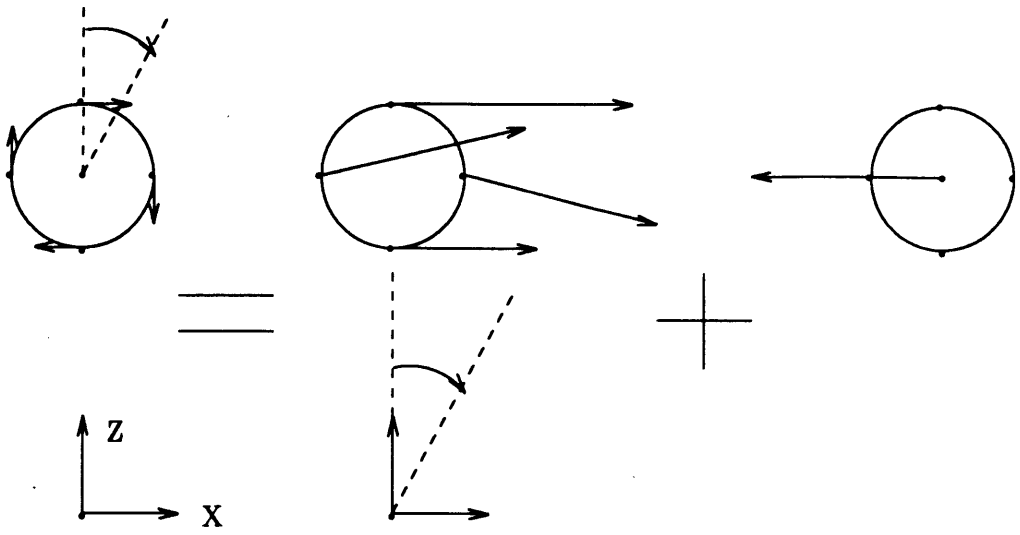

Figure 4-2: A general small motion (here, a rotation about a body axis) is equal to the sum of a rotation and translation relative to the space axes. If the motion is infinitesimal it may be represented as a pair of vectors.

coordinates is sufficiently small that a vector representation is possible [Goldstein, 1950] (Figure 4-2).

In the case of a rigid body moving relative to the space set of axes with rotational component $\mathbf{\Omega}$ and translational component $\mathbf{T}$, all points on the body have a motion given by

$$\frac{d\mathbf{R}}{dt} = -\mathbf{\Omega} \times \mathbf{R} - \mathbf{T}. \tag{4.6}$$

With substitution for $\mathbf{R}$, this may be rewritten as

$$\frac{d\mathbf{R}}{dt} = -\frac{\mathbf{R} \cdot \hat{\mathbf{z}}}{F} \mathbf{\Omega} \times \mathbf{r} - \mathbf{T}. \tag{4.7}$$

The equation for optical flow of a rigid body may be obtained by substituting this expression for $d\mathbf{R}/dt$ into the formula derived earlier for $d\mathbf{r}/dt$:

$$\frac{d\mathbf{r}}{dt} = -\left(\hat{\mathbf{z}} \times \left(\mathbf{r} \times \left(\frac{1}{F}\mathbf{r} \times \boldsymbol{\Omega} - \frac{1}{\mathbf{R} \cdot \hat{\mathbf{z}}}\mathbf{T}\right)\right)\right). \tag{4.8}$$

Recovering $\boldsymbol{\Omega}$ and $\mathbf{T}$ will not require explicit calculation of the optical flow, however, as will be shown below.

4.4 The brightness constancy constraint

Another constraint to be placed upon the problem formulation involves image brightness. In an environment with shadow regions or strongly directional lighting, brightness may be very dependent upon position or orientation. With fairly diffuse lighting that varies only slowly with time, and for small motions, the brightness of an image patch remains approximately constant as the patch moves:

$$\frac{dI}{dt} = 0, \tag{4.9}$$

or

$$\frac{\partial I}{\partial \mathbf{r}} \cdot \frac{d\mathbf{r}}{dt} + \frac{\partial I}{\partial t} = 0: \tag{4.10}$$

Here the image brightness gradient is

$$\frac{\partial I}{\partial \mathbf{r}} = \begin{pmatrix} \frac{\partial I}{\partial x} \\ \frac{\partial I}{\partial y} \\ 0 \end{pmatrix}. \tag{4.11}$$

Substituting the rigid body expression for $d\mathbf{r}/dt$ gives the constraint equation

$$\frac{\partial I}{\partial t} - \frac{\partial I}{\partial \mathbf{r}} \cdot \left(\hat{\mathbf{z}} \times \left(\mathbf{r} \times \left(\frac{1}{F} \mathbf{r} \times \mathbf{\Omega} - \frac{1}{\mathbf{R} \cdot \hat{\mathbf{z}}} \mathbf{T} \right) \right) \right) = 0. \tag{4.12}$$

This may be simplified by defining two vectors: $\mathbf{v} = -(1/F)((\partial I/\partial \mathbf{r} \times \hat{\mathbf{z}}) \times \mathbf{r}) \times \mathbf{r}$, and $\mathbf{s} = (\partial I/\partial \mathbf{r} \times \hat{\mathbf{z}}) \times \mathbf{r}$:

$$\frac{\partial I}{\partial t} + \mathbf{v} \cdot \mathbf{\Omega} + \frac{1}{\mathbf{R} \cdot \hat{\mathbf{z}}} \mathbf{s} \cdot \mathbf{T} = 0, \tag{4.13}$$

which is the final constraint equation, expressed in terms of the spatiotemporal image gradients and z values. The elements of the $\mathbf{s}$ and $\mathbf{v}$ vectors are

$$\mathbf{s} = \begin{pmatrix} -F \frac{\partial I}{\partial x} \\ -F \frac{\partial I}{\partial y} \\ y \frac{\partial I}{\partial y} + x \frac{\partial I}{\partial x} \end{pmatrix} \quad \mathbf{v} = \begin{pmatrix} (F + \frac{y^2}{F}) \frac{\partial I}{\partial y} + \frac{xy}{F} \frac{\partial I}{\partial x} \\ (-F - \frac{x^2}{F}) \frac{\partial I}{\partial x} - \frac{xy}{F} \frac{\partial I}{\partial y} \\ y \frac{\partial I}{\partial x} - x \frac{\partial I}{\partial y} \end{pmatrix}. \tag{4.14}$$

In the following section the estimation of the gradients necessary to solve the constraint equation will be discussed.

4.5 Signal estimation to compute spatiotemporal gradients

A significant problem with spatiotemporal constraint methods is the requirement for accurate estimates of spatiotemporal image gradients. The gradients of a pair of input images can be approximated by finite difference methods if the signal is bandlimited and noiseless, and if the images are spatially and temporally sampled at high enough rates. Real images, though, contain significant additive noise, which is enhanced by simple differentiation schemes [Horn, 1986]; also, there is often aliasing, particularly in the temporal direction, which can produce incorrect results with local finite-difference methods. An approach suggested by Martinez and Krause [Martinez, 1986][Krause, 1987][5] involves parametric signal estimation of the form

$$I(x,y,t) \approx \tilde{I}(x,y,t) = \sum_{i=1}^{N} S_i \phi_i(x,y,t). \tag{4.15}$$

Sample intensity values from the input images are used to estimate the model parameters S_i, and the gradients obtained through partial differentiation of $\tilde{I}(x,y,t)$ in terms of x, y, and t. The signal model itself is specified through appropriate choice of the

[5] Only Martinez's signal model – not his motion estimator – will be employed in this work.

basis functions ϕ_i. Martinez proposes a basis set of three-dimensional polynomials:

$$\begin{aligned} \phi_1 &= 1 \quad \phi_2 = x \quad \phi_3 = y \\ \phi_4 &= t \quad \phi_5 = x^2 \quad \phi_6 = y^2 \\ \phi_7 &= xy \quad \phi_8 = xt \quad \phi_9 = yt \end{aligned} \tag{4.16}$$

The one-dimensional matrix of coefficients $\mathbf{S}$ is calculated by minimizing the error between the intensity values observed and those predicted by the model, specifically

$$\min \sum_i (I(x_i, y_i, t_i) - \mathbf{S}\Phi(x_i, y_i, t_i))^2. \tag{4.17}$$

Typically a small spatiotemporal region will be modeled (say 5-by-5 pixels from two adjacent frames – a larger spatial region will permit more accurate measurement of larger motions, but at the expense of additional computation), resulting in an overconstrained set of linear equations: 50 samples to estimate 9 parameters. The parameter vector $\mathbf{S}$ may be expressed as

$$\mathbf{S} = \mathbf{QI}, \tag{4.18}$$

where

$$\mathbf{I} = \begin{pmatrix} I(x_1, y_1, t_1) \\ \vdots \\ I(x_{50}, y_{50}, t_{50}) \end{pmatrix} \tag{4.19}$$

and $\mathbf{Q}$ is the 9×50 pseudoinverse of a 50×9 matrix $\mathbf{A}$

$$\mathbf{Q} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T. \tag{4.20}$$

More detail on solving large systems of linear equations of this sort may be found in numerical methods references such as [Constantinides, 1987] or [Kohn, 1987].

The elements of $\mathbf{A}$ are the basis functions evaluated at each sample point:

$$\mathbf{A} = \begin{pmatrix} 1 & x_1 & y_1 & t_1 & x_1^2 & y_1^2 & x_1y_1 & x_1t_1 & y_1t_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{50} & y_{50} & t_{50} & x_{50}^2 & y_{50}^2 & x_{50}y_{50} & x_{50}t_{50} & y_{50}t_{50} \end{pmatrix}. \tag{4.21}$$

Computation is reduced by re-origining the center of each 5-by-5 image window to $(0,0)$ and declaring the t values of the two frames to be 0 and 1. Thus the very expensive calculation of $\mathbf{Q}$ need be carried out only once, as it is independent of the observed intensity values, and only the multiplication of $\mathbf{Q}$ and $\mathbf{I}$ must be performed for each pair of sample windows. Once the resulting $\mathbf{S}$ is found, the gradients at the center of the spatiotemporal sample block $(x, y, t) = (0, 0, 1/2)$ are straightforwardly computed:

$$\frac{\partial I}{\partial x} = \sum_{i=1}^{9} S_i \left.\frac{\partial \phi_i}{\partial x}\right|_{(x=0,y=0,t=\frac{1}{2})} = S_2 + \frac{1}{2}S_8. \tag{4.22}$$

In a like fashion

$$\frac{\partial I}{\partial y} = S_3 + \frac{1}{2}S_9 \tag{4.23}$$

and

$$\frac{\partial I}{\partial t} = S_4. \tag{4.24}$$

It is important to note that the final solutions are not functions of a number of elements of $\mathbf{S}$, specifically S_1, S_5, S_6, and S_7. This does not mean that the corresponding basis polynomials could be eliminated from the signal model. Doing so would cause the values of the desired S_i to change in an attempt to account for the additional error that would result from the omission of these model parameters. Thus the full system of equations must be solved in order that the derivatives will take on the correct values for this model. The partial derivatives of intensity with respect to time, x, and y computed for a pair of images of a moving scene are shown in Figures 4-3, 4-4, 4-5, and 4-6.

4.6 Least-squares solution of the constraint equation

While it should be possible to recover the six unknowns $\Omega_x, \Omega_y, \Omega_z, T_x, T_y$, and T_z by using the range data and intensity gradients at just a few image points, more accurate and robust velocity estimates will result from using a least-squares error minimization over an entire rigid object. The method presented here is similar to that proposed by

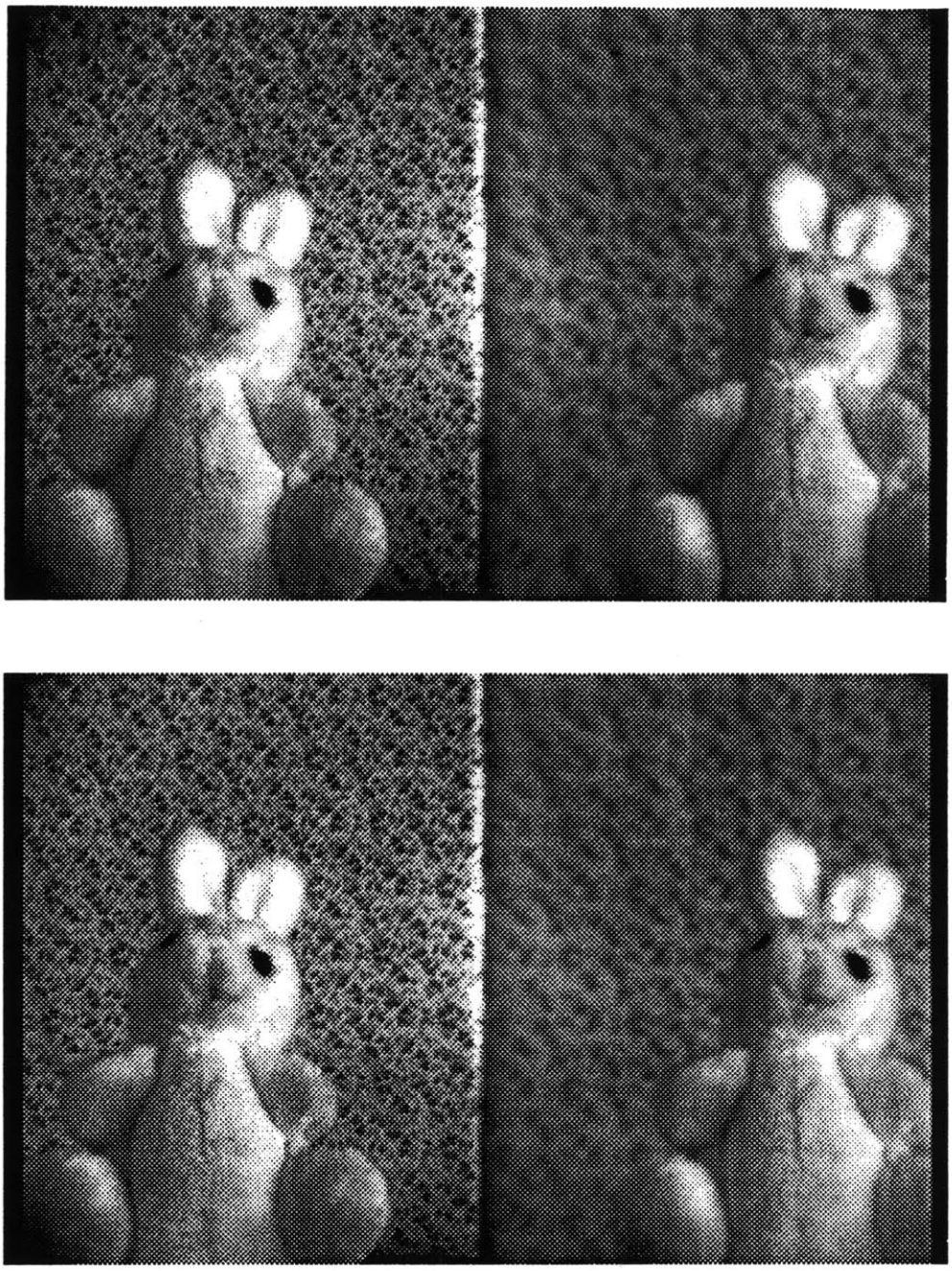

Figure 4-3: A pair of frames of a rotating object.

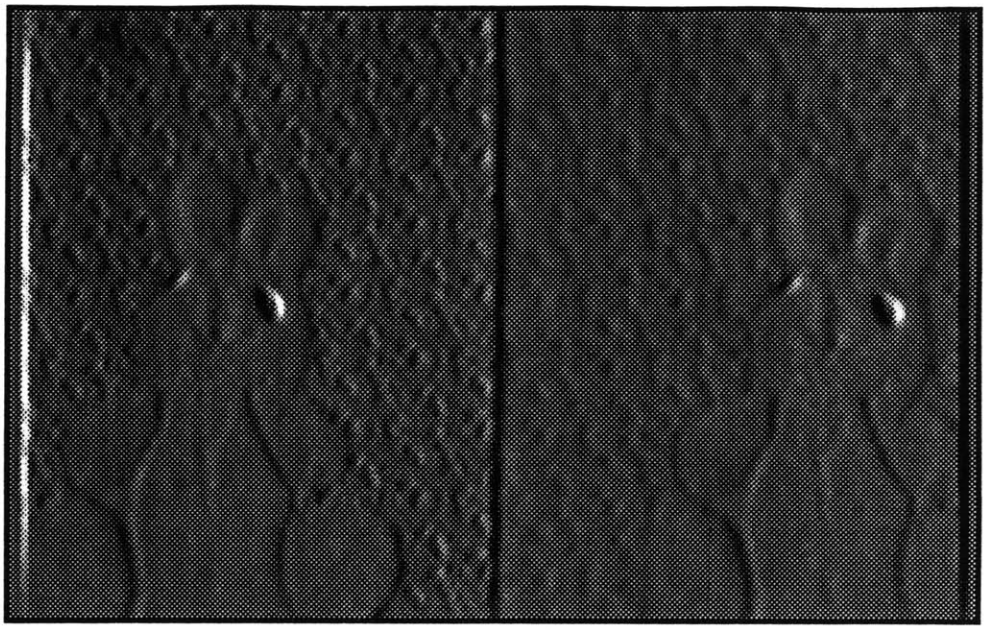

Figure 4-4: Partial derivative of intensity with respect to x for Figure 4-3.

Negahdaripour and Horn [Negahdaripour, 1987] (though the referenced paper solves only for the case of a planar surface); a more complete discussion of least-squares minimization for motion recovery may be found in [Horn, 1986].

It is desired to compute the motion components $\mathbf{\Omega}$ and $\mathbf{T}$ which minimize the integral

$$f = \int\int \left(\frac{\partial I}{\partial t} + \mathbf{v} \cdot \mathbf{\Omega} + \frac{1}{\mathbf{R} \cdot \hat{\mathbf{z}}} \mathbf{s} \cdot \mathbf{T} \right)^2 dx\, dy \tag{4.25}$$

over some region representing a rigid object. Differentiating f with respect to $\mathbf{\Omega}$ and

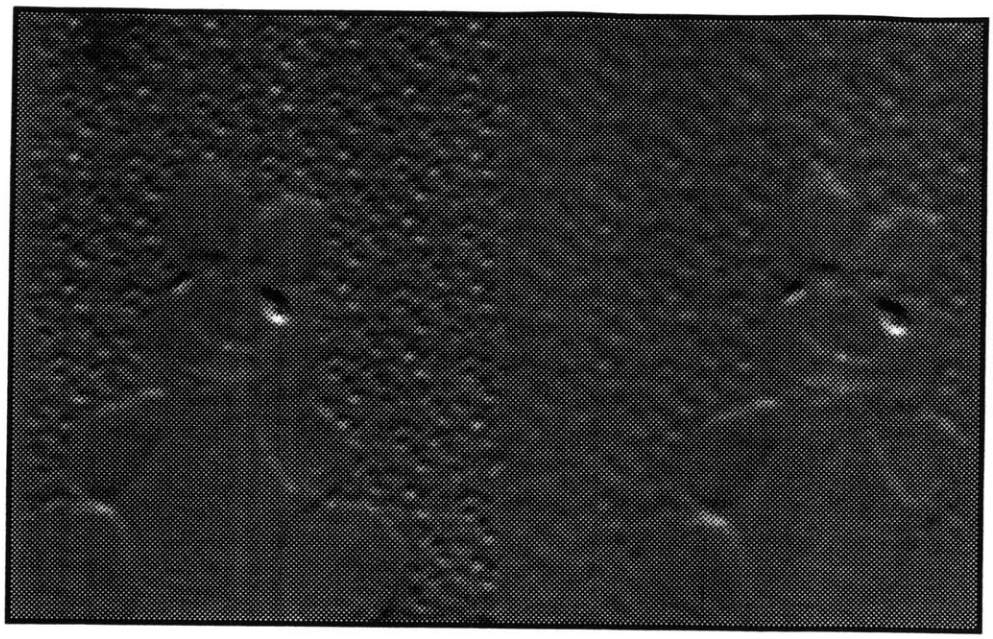

Figure 4-5: Partial derivative of intensity with respect to y for Figure 4-3.

$\mathbf{T}$ and setting the results to zero yields a pair of equations

$$\int\int \left(\frac{\partial I}{\partial t} + \mathbf{v} \cdot \mathbf{\Omega} + \frac{1}{\mathbf{R} \cdot \hat{\mathbf{z}}} \mathbf{s} \cdot \mathbf{T} \right) \mathbf{v} \, dx \, dy = 0 \tag{4.26}$$

$$\int\int \left(\frac{\partial I}{\partial t} + \mathbf{v} \cdot \mathbf{\Omega} + \frac{1}{\mathbf{R} \cdot \hat{\mathbf{z}}} \mathbf{s} \cdot \mathbf{T} \right) \frac{1}{\mathbf{R} \cdot \hat{\mathbf{z}}} \mathbf{s} \, dx \, dy = 0. \tag{4.27}$$

These equations may be rearranged

$$\mathbf{\Omega} \int\int (\mathbf{v}\mathbf{v}^T) \, dx \, dy + \mathbf{T} \int\int \frac{1}{\mathbf{R} \cdot \hat{\mathbf{z}}} (\mathbf{v}\mathbf{s}^T) \, dx \, dy = - \int\int \frac{\partial I}{\partial t} \mathbf{v} \, dx \, dy \tag{4.28}$$

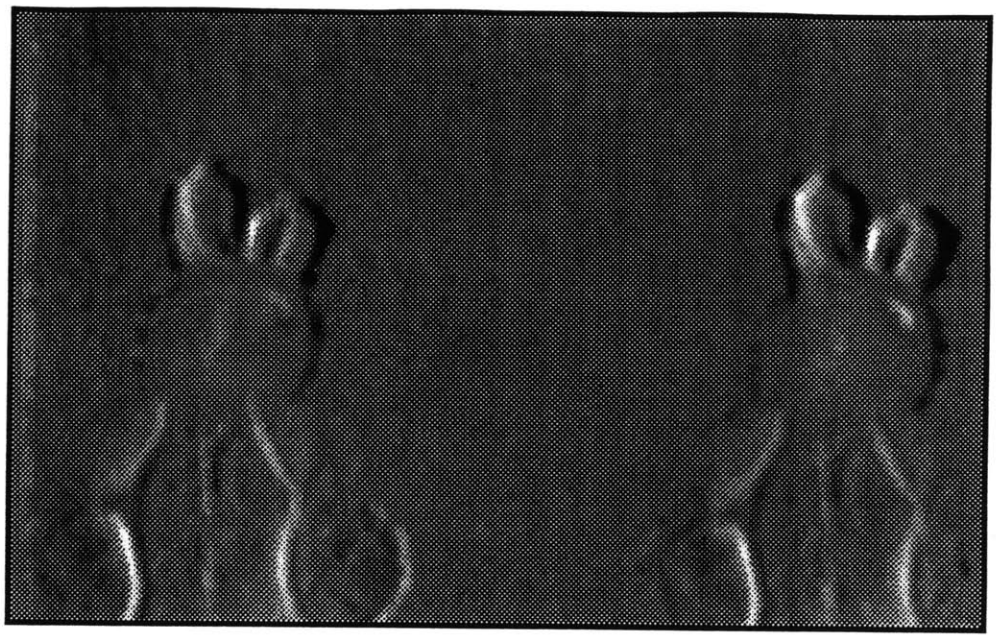

Figure 4-6: Partial derivative of intensity with respect to t for Figure 4-3.

$$\boldsymbol{\Omega} \int\int \frac{1}{\mathbf{R}\cdot\hat{\mathbf{z}}}(\mathbf{s}\mathbf{v}^T)\,dx\,dy + \mathbf{T}\int\int \frac{1}{(\mathbf{R}\cdot\hat{\mathbf{z}})^2}(\mathbf{s}\mathbf{s}^T)\,dx\,dy = -\int\int \frac{\partial I}{\partial t}\frac{1}{\mathbf{R}\cdot\hat{\mathbf{z}}}\mathbf{s}\,dx\,dy, \quad (4.29)$$

which suggests the following grouping:

$$\begin{pmatrix} \mathbf{M}_1 & \mathbf{M}_2 \\ \mathbf{M}_3 & \mathbf{M}_4 \end{pmatrix} \begin{pmatrix} \boldsymbol{\Omega} \\ \mathbf{T} \end{pmatrix} = - \begin{pmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \end{pmatrix}, \quad (4.30)$$

where

$$\mathbf{M}_1 = \int\int (\mathbf{v}\mathbf{v}^T)\,dx\,dy \quad (4.31)$$

$$\mathbf{M}_2 = \int\int \frac{1}{\mathbf{R}\cdot\hat{\mathbf{z}}}(\mathbf{v}\mathbf{s}^T)\,dx\,dy \quad (4.32)$$

$$\mathbf{M}_3 = \int\int \frac{1}{\mathbf{R}\cdot\hat{\mathbf{z}}}(\mathbf{s}\mathbf{v}^T)\,dx\,dy = \mathbf{M}_2^T \tag{4.33}$$

$$\mathbf{M}_4 = \int\int \frac{1}{(\mathbf{R}\cdot\hat{\mathbf{z}})^2}(\mathbf{s}\mathbf{s}^T)\,dx\,dy \tag{4.34}$$

$$\mathbf{d}_1 = \int\int \frac{\partial I}{\partial t}\mathbf{v}\,dx\,dy \tag{4.35}$$

$$\mathbf{d}_2 = \int\int \frac{\partial I}{\partial t}\frac{1}{\mathbf{R}\cdot\hat{\mathbf{z}}}\mathbf{s}\,dx\,dy. \tag{4.36}$$

The solutions to this system may be expressed in numerous ways; one expression is given by

$$\mathbf{T} = (\mathbf{M}_4 - \mathbf{M}_2^T\mathbf{M}_1^{-1}\mathbf{M}_2)^{-1}(\mathbf{M}_2^T\mathbf{M}_1^{-1}\mathbf{d}_1 - \mathbf{d}_2) \tag{4.37}$$

$$\mathbf{\Omega} = -\mathbf{M}_1^{-1}(\mathbf{d}_1 + \mathbf{M}_2\mathbf{T}). \tag{4.38}$$

If this technique is being employed throughout an extended sequence, an additional assumption may be applied in order to increase the accuracy of the recovered motions. Inasmuch as the objects in the scene should have significant inertia, the magnitudes and directions of their motions should not change drastically from one increment of time to the next – in other words, the motion should be locally smooth in the temporal domain. An easy way to exploit this phenomenon is to apply a smoothing filter across sequences of motion estimates. A Gaussian filter with a σ of four frames has been used for this purpose in the research described herein, but there has been no in-depth investigation as to what the characteristics of an optimal temporal smoothing filter should be.

It was noted in the preceding chapter that depth-from-focus will perform better given images which represent the state of the scene at a particular instant rather than integrated over a frame time. Eliminating motion blur will, on the other hand, open up the possibility of significant temporal aliasing leading to incorrect motion estimation. Were motion estimation computed locally, this would be a major concern, but here the equations are being integrated over a fairly large region representing one rigid scene element and local errors should not greatly affect the outcome. It is additionally desirable that the surface of the object model to be built up be as unblurred as possible. On balance, therefore, a shuttered video camera will probably produce better results overall within the context of the requirements of this project.

Most interlaced video cameras with shutters open the shutter at the beginning of each field, not each frame. When such a camera is used, it is possible – if the scene does not contain too much vertical detail – to pretend that the camera is really progressively-scanned with half the number of lines. In this case, the motion will be measured between adjacent fields rather than frames, effectively doubling the fastest motion that may be accurately estimated for a given size of spatiotemporal sampling block in the signal estimator.

4.7 Improving range estimates with the constraint equation

An interesting extension to the motion recovery process is the use of the constraint equation and the motion solutions derived above to improve the range data once motion has been estimated. The range, of course, varies from point to point across a rigid object, so error minimization with respect to range over some region is not useful here. A more fruitful approach involves surface orientation. If the object is assumed to be locally smooth, small surface patches may be approximated as planes. Mathematically,

$$\mathbf{R} \cdot \mathbf{n} = 1, \tag{4.39}$$

where $\mathbf{n}$ is an "inward directed" normal vector to the plane. Since $\mathbf{R} = \mathbf{r}(\mathbf{R} \cdot \hat{\mathbf{z}})/F$ this may be rewritten:

$$\mathbf{r} \cdot \mathbf{n} = \frac{F}{\mathbf{R} \cdot \hat{\mathbf{z}}}. \tag{4.40}$$

Now it is desired to minimize over some small patch of an object

$$f = \int\int \left(\frac{\partial I}{\partial t} + \mathbf{v} \cdot \mathbf{\Omega} + \frac{1}{F}(\mathbf{r} \cdot \mathbf{n})(\mathbf{s} \cdot \mathbf{T}) \right)^2 dx\, dy. \tag{4.41}$$

Differentiating with respect to $\mathbf{n}$ and setting equal to zero provides the system of linear equations

$$\int\int \left(\frac{\partial I}{\partial t} + \mathbf{v} \cdot \mathbf{\Omega} + \frac{1}{F}(\mathbf{r} \cdot \mathbf{n})(\mathbf{s} \cdot \mathbf{T}) \right) \frac{1}{F}(\mathbf{s} \cdot \mathbf{T})\mathbf{r} \, dx \, dy = 0. \tag{4.42}$$

Rearranging,

$$\mathbf{n} \int\int \frac{(\mathbf{s} \cdot \mathbf{T})^2}{F^2}(\mathbf{r}\mathbf{r}^T) \, dx \, dy = -\int\int \left(\frac{\partial I}{\partial t} + (\mathbf{v} \cdot \mathbf{\Omega}) \right) \frac{(\mathbf{s} \cdot \mathbf{T})}{F}\mathbf{r} \, dx \, dy. \tag{4.43}$$

The solution is simply

$$\mathbf{n} = -\left(\int\int \frac{(\mathbf{s} \cdot \mathbf{T})^2}{F^2}(\mathbf{r}\mathbf{r}^T) \, dx \, dy \right)^{-1} \left(\int\int \left(\frac{\partial I}{\partial t} + (\mathbf{v} \cdot \mathbf{\Omega}) \right) \frac{(\mathbf{s} \cdot \mathbf{T})}{F}\mathbf{r} \, dx \, dy \right), \tag{4.44}$$

from which

$$\mathbf{R} \cdot \hat{\mathbf{z}} = \frac{F}{\mathbf{r} \cdot \mathbf{n}}. \tag{4.45}$$

Direct application of the above method produces range data containing a small amount of noise in the form of grossly incorrect (approaching zero or infinity) distance values. These turn out to correspond to surface normals which lie entirely in the image plane or point toward the camera, owing to occasional areas where noise or modeling error causes local intensity gradients inconsistent with the motion of the body as a whole. As the vector $\mathbf{n}$ has been defined to be inward-pointing, there should be no visible regions with normals toward the camera. Again assuming smoothness, range values for locations with obviously misdirected normals may be assigned by a weighted

average of neighboring points.

A good minimization region size has proven to be 5-by-5 or 7-by-7 pixels, larger patches over-smoothing the object surface in addition to being more computationally expensive.

The "new" set of range values computed in this manner will presumably be combined with the depth-from-focus range estimates rather than used to replace them. A simple way to do this is a weighted average, but a better and smoother surface results from a minimization solution of the sort discussed in Section 3.1, though here there will be two sets of original z values for which the surface must minimize mean-squared error. Even starting from a fairly blocky depth-from-focus range image, the result (Figure 4-7) begins to approach the quality of the laser rangefinder data from Chapter 2. More careful analysis suggests that the process should operate in a way that exploits the particular characteristics of each rangefinding process. Specifically, as depth-from-focus is known to give reasonably accurate absolute range values but limited (x, y) resolution, while structure-from-motion gives better local detail, the former's low-spatial-frequency components and the latter's high-spatial-frequency components should be given more priority.

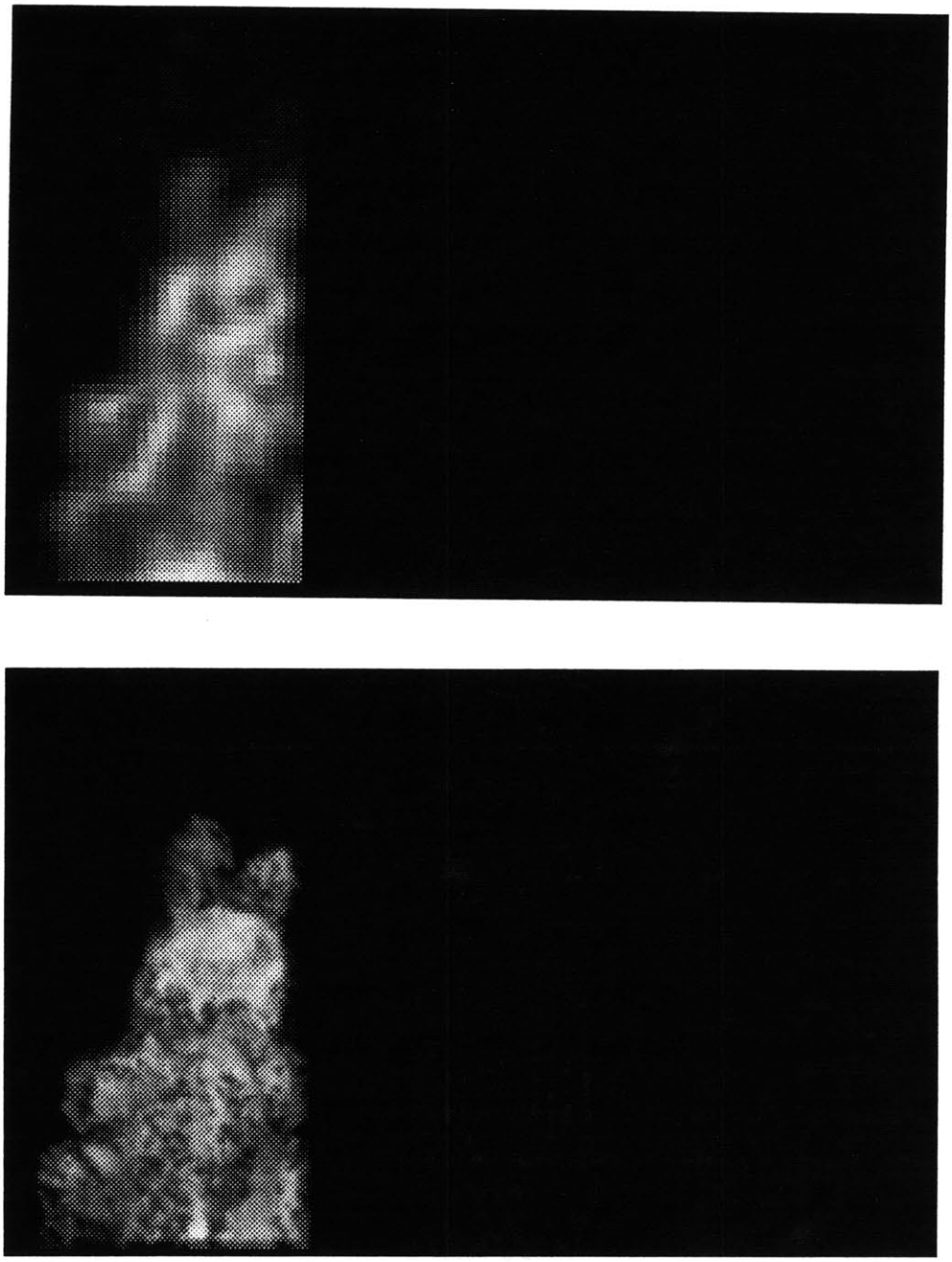

Figure 4-7: Blocky depth-from-focus image (top) is improved through structure-from-motion (bottom).

4.8 Range sensitivity as a function of motion direction

It should be apparent that the quality of the range estimates from the motion estimator will vary with the characteristics of the scene motion. In general, larger motions (as long as they are not so large that they overwhelm the gradient estimator) would be supposed to produce better range detail, owing to the increased variation in the two-dimensional projection of the motion as a function of three-dimensional position. But an examination of the optical flow equations from sections 4.2 and 4.3 will show that not all directions of motion produce equal flow for a given displacement.

It will be recalled that for a rigid body

$$\frac{d\mathbf{r}}{dt} = -\left(\hat{\mathbf{z}} \times \left(\mathbf{r} \times \left(\frac{1}{F}\mathbf{r} \times \boldsymbol{\Omega} - \frac{1}{\mathbf{R}\cdot\hat{\mathbf{z}}}\mathbf{T}\right)\right)\right). \tag{4.46}$$

Expanding this expression and recasting it in terms of the world coordinates of a point $\mathbf{R}$ yields a less neat but more illuminating result:

$$\begin{aligned}\frac{d\mathbf{r}}{dt} = & -\frac{F}{Z}\left(\frac{XY}{Z}\Omega_x - \left(\frac{X^2}{Z}+F\right)\Omega_y + Y\Omega_z - T_x - \frac{X}{Z}T_z\right)\hat{\mathbf{x}} \\ & -\frac{F}{Z}\left(\left(\frac{Y^2}{Z}+F\right)\Omega_x - \frac{XY}{Z}\Omega_y - X\Omega_z - T_y - \frac{Y}{Z}T_z\right)\hat{\mathbf{y}}.\end{aligned} \tag{4.47}$$

A number of inferences may be made from this equation; one of particular interest

concerns translatory motion. Considering just the x component for a moment (y being analogous), the flow from a translation in the x direction is greater than that from an equal translation in the z direction by the factor Z/X. Thus for points less than 45 degrees away from the optical axis, translation in x or y is better for structure-from-motion purposes than z translation.[6] Indeed, making effective use of motions which consist largely of translations in z would require a fairly wide-angle lens. Also by inspection, rotation about z will not give good range information for points close to the optical axis.

As in Section 3.7, the calculus of errors may be applied to this expression. For clarity and compactness, only the x component of flow for a strictly translational motion will be considered in this analysis. Therefore,

$$\frac{Z^2}{F}\frac{d\mathbf{r}}{dt} - ZT_x - XT_z = 0 \tag{4.48}$$

and

$$Z = \frac{F}{2\frac{d\mathbf{r}}{dt}}\left(T_x \pm \sqrt{T_x^2 - \frac{4XT_z}{F}\frac{d\mathbf{r}}{dt}}\right). \tag{4.49}$$

Replacing $d\mathbf{r}/dt$ by $(d\mathbf{r}/dt + \epsilon_d)$, the error in Z as a function of the error in measuring

[6] Matthies, *et al.* reach the same conclusion, via somewhat different reasoning [Matthies, 1987].

optical flow on the image plane is

$$\epsilon_Z = \left(-\frac{Z}{\left(\frac{dr}{dt} + \epsilon_d\right)^2} \pm \frac{XT_z}{\left(\frac{dr}{dt} + \epsilon_d\right)\sqrt{T_x^2 - \frac{4XT_x}{F}\left(\frac{dr}{dt} + \epsilon_d\right)}} \right) \epsilon_d, \tag{4.50}$$

an expression which suggests that the error actually decreases at a faster than linear rate as the optical flow increases.

4.9 Chapter summary

To review, the following assumptions have been made regarding the scene elements and the imaging process:

- The lens distortion is insignificant, and a pinhole camera model will suffice.
- Object motions are infinitesimal, and may be expressed as vectors.
- It is possible to segment the scene into rigid pieces.
- Object surfaces are locally smooth.
- Objects are diffusely reflecting.
- Scene lighting is diffuse, and varies slowly as a function of position and time.
- Inertia will limit changes in object motions, implying that motion will be locally smooth temporally.

It has been shown that application of these assumptions permits computing incremental frame-time-to-frame-time motion for each rigid piece. Depth-from-focus data is used along with spatiotemporal image gradients from a parametric signal model to drive a three-dimensional motion estimator; assuming the motions thus calculated and running the process essentially in reverse gives structure-from-motion range constraints which refine the input range data. The information thus obtained will be put to use in the next chapter.

Chapter 5

Building object models

Di quest'onda che rifluisce dai recordi la città s'imbeve come una spugna e si dilata. Una descrizione di Zaira quale è oggi dovrebbe contenere tutto il passato di Zaira.

As this wave flows in from memory, the city soaks it up like a sponge and expands. A description of Zaira as it is today should contain all Zaira's past.

Italo Calvino
Le città invisibili

5.1 Goals

At this point in the processing of an input image sequence, the following information has been extracted: motion-improved range information for each input frame (which can be combined with the intensity information and placed into particle form), object segmentation for each frame, and incremental motion estimates describing the translation and rotation of each object from one frame time to the next. It is readily apparent

that this representation of a moving scene is highly redundant and should allow for much data compresssion – given the motions which transform the scene as observed at one instant to that at the next instant, storing a description of the scene at the second instant seems unnecessary. The only obstacle to implementing this scheme is the 2 1/2-D nature of each frame's description. As an object moves, it also uncovers new points, and these will be missing if a 2 1/2-D description is translated and rotated. A more useful representation will result if each input frame contributes its "new" points to the building up of a volumetric 3-D scene description. Reconstructing the original movie from this database will enable re-rendering from different viewpoints without missing parts, as long as the parts in question were seen by the camera at some point in the sequence.

5.2 Related work

As noted earlier, computer graphics commonly employs object descriptions such as meshes of polygons which simplify calculation, and – with the exception of fractal curves [Mandelbrot, 1983] – no claim is generally made of the "naturalness" of such descriptions. Polygons are attractive for some industrial vision applications as well, since (once the observed scene has been "polygonalized") they permit computations to be performed only at vertex points; also in many situations in which such systems are applicable the objects to be recognized are easily described in polygonal form.

Researchers interested in modeling biological vision systems, on the other hand, prefer object descriptions which are more conceptually understandable, and which make clear the articulation of objects into parts. The idea of the "generalized cylinder"[1] was introduced by Binford, and represents volumetric shapes as the result of moving a size-but-not-shape-varying cross-section along a (possibly curved) axis [Binford, 1971]. This concept was later employed by Marr and Nishihara in a hierarchical system for creating a part-based conceptual description of objects [Marr, 1982]. Pentland used the superquadric – a modification of the equation for the sphere – and created a process-oriented object editor in which object models were created by deformation of "balls of clay," as well as a method for fitting superquadric models to range data [Pentland, 1987a]. Terzopoulos, *et al.* extended the energy-minimizing splines called "snakes" to three dimensions, such that a quasi-symmetrical volume, not just a 2-D curve, is fitted to image contours [Terzopoulos, 1987].

Besides being more or less restricted in the imagery which can be represented easily,[2] the parametric descriptions of the preceding paragraph substitute an assumption of symmetry (or quasi-symmetry) or regularity for explicit knowledge of unseen portions of the object. While this assumption may be a desirable property in many cases, it does not necessarily provide for easy refinement of the model when additional information in the form of different views of an object becomes available. Apart from Linhardt's

[1]This term is used interchangeably with "generalized *cone*."

[2]Pentland expanded the possible objects which could be represented in his system by allowing surface properties and Boolean operations (most importantly, NOT) to take part in the descriptions.

earlier-cited work, which employed the same particle descriptions outlined in Chapter 2, research upon building up object models from multiple range views of a scene has generally employed polygons. Polygon models permit relatively simple addition of newly-seen points, as the inclusion of new information will have mostly local effects and should not require total recomputation of the object model. Boissonat and Faugeras describe a technique for representing objects by constructing a polyhedral representation from range points, and suggest that these may come from more than one view of an object [Boissonat, 1981]. An extension to their work is provided by Bhanu, who takes 3-D points from several range views of an object and segments the object into planar faces; he, like Linhardt, assumes that the rotation angle of the object relative to the camera is known for each view, but once the full model has been built up the position of a new, unknown view is found by comparing it with the object description [Bhanu, 1984].

A completely different approach to the model-building problem, and one much closer to the spirit of this thesis, is found in the work of Kappei and Liedtke [Kappei, 1987], though they employ only intensity images to drive their process. They begin with a polygonal approximation to an object model based upon *a priori* knowledge or upon rotating a silhouette around an axis. Iterative motion estimation is performed upon the image regions corresponding to the neighborhoods of the vertex points and the vertices are allowed to move along the line of sight from the camera to minimize error in the motion estimator, thus improving the model.

5.3 Approach

A reasonable approach to this reconstruction task would involve using the first instance of an object to define a set of body coordinate axes coincident with the global (likely camera-centered) axes.[3] Range and intensity values from later observations of the object would be rotated and translated back into the coordinate system of these original axes – according to the product of incremental estimated motions up till that later time – and assembled into a particle database with redundant particles eliminated to keep the object description as compact as possible. The final outcome of processing an input motion sequence in this manner is a three-dimensional description of the surfaces of objects, and of their motions through time. Obviously, these motion vectors can be interpolated, and the locations, orientations, and appearances of objects can be reconstructed for instants other than those at which the camera sampled the scene.

A schematic of this process is shown in Figure 5-1. In practice, adding an additional degree of intelligence to the database building software greatly improves the accuracy of the resulting databases and keeps their size within reasonable bounds even for long input sequences. While the motion estimator error for any particular pair of input frames may be small, after many dozens of frames the product of incremental errors builds up to the point where it may affect the addition of new points to the database.

[3] If an object is known – or observed – to have a "natural" set of three-dimensional body axes the use of which would simplify subsequent calculations, these could be applied instead with no loss of generality.

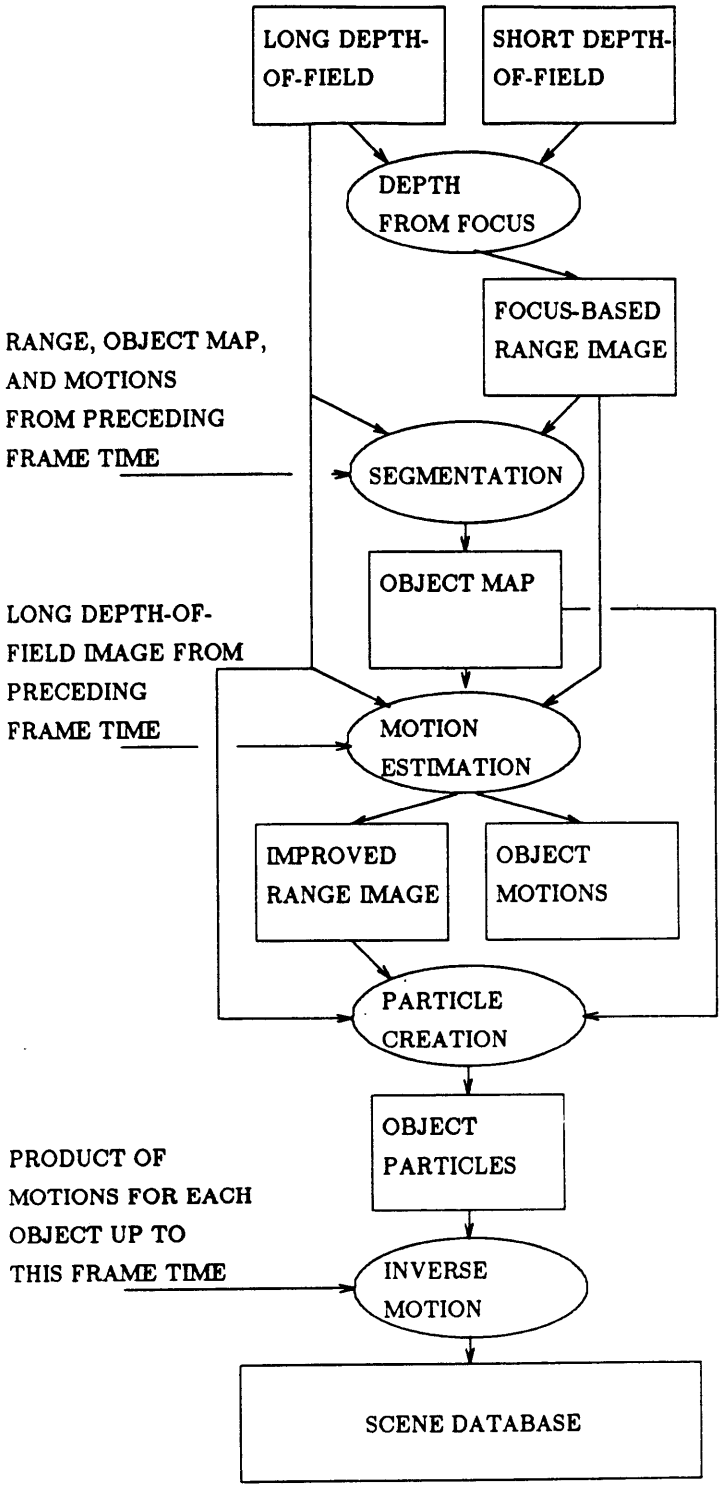

Figure 5-1: Schematic representation of the model-building process.

For an object with a significant rotational component about the x or y axes,[4] the database building software attempts to line up the silhouette of the "trailing" edge of an input view with the corresponding region of the database, as well as making sure that the intensities of the points already existing in the database match fairly well those that the input view places in the same locations.[5] The position error term is retained so as to assist in the incremental-motion-to-absolute-position conversion of later input frames. Also, new points are added only in regions of space that are relatively unpopulated (which for the rotating object above would be the "leading" edge, while in the case of a background object would be those portions that a moving foreground object had previously occluded).

An additional optimization should be mentioned at this point. Readers who have been following the discussion for the past several chapters may be wondering how, given an admittedly imperfect scene segmentation algorithm and occasionally erroneous range information, "foreground" particles can be kept out of the "background" database and vice versa. The answer to the dilemma lies in making the segmenter very conservative; ambiguous image regions in a given input frame which might be assigned to one object or another are actually assigned to neither in the database building process. At some later time in the input sequence, these points will be presumably be seen again (with less ambiguity) and may then be incorporated into the database.

[4]Apart from the effects of perspective, z-axis rotation of an object not partially occluded by another closer to the lens generally does not uncover a large number of new points.

[5]This foreshadows the possibility of matching a pre-existing object model with the input views as a check on the motion estimator, an extension which will be suggested in the following chapter.

5.4 Evaluation

Two different image sequences have been recorded on 1" videotape, digitized frame-by-frame, and processed in this manner. In the first, the motion of a rotating pendulum with patterned surface (actually a rubber ball suspended by a wire) was tracked, and the new particles from each frame time were added to a particle database. Figures 5-2 and 5-3 illustrate how the model of the object builds up throughout the sequence.

In a second sequence (Figure 5-4), the author of this thesis moved across the camera's field of view, made a roughly 270-degree turn, turned back, and bent forward at the waist, with the camera viewing the figure from the waist upwards. Arms and head were not moved independently of the torso. The range data from the scene, and thus the resulting database, were segmented into two objects: the figure and the wall behind him. As all parts of the wall were at some time unoccluded from the camera's viewpoint, it is possible to build up a complete model of the wall by processing the entire sequence; likewise a relatively complete model of the figure should be possible given that the majority of the figure's surface faced the camera at some point, and that every portion was within 45 degrees of the line of sight.

Because of the length of this sequence (some 900 video frames) and the slowness of the motion, calculation was speeded up by skipping many of the frames in the database-building process (though motion estimation must be performed for every frame time to enable calculating absolute position). This is in contrast to the spinning ball sequence,

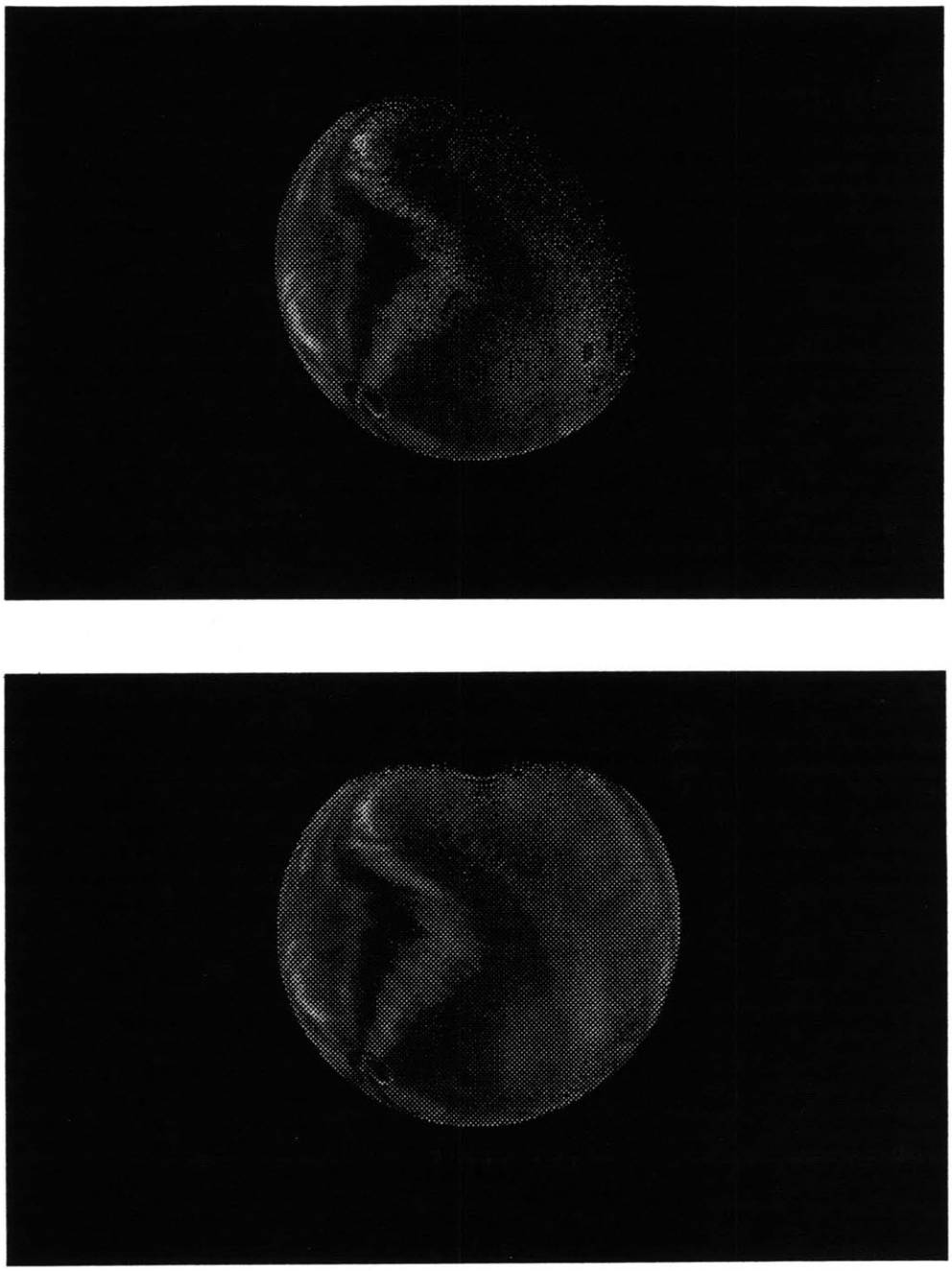

Figure 5-2: A database builds up over time as the camera observes new parts of an object.

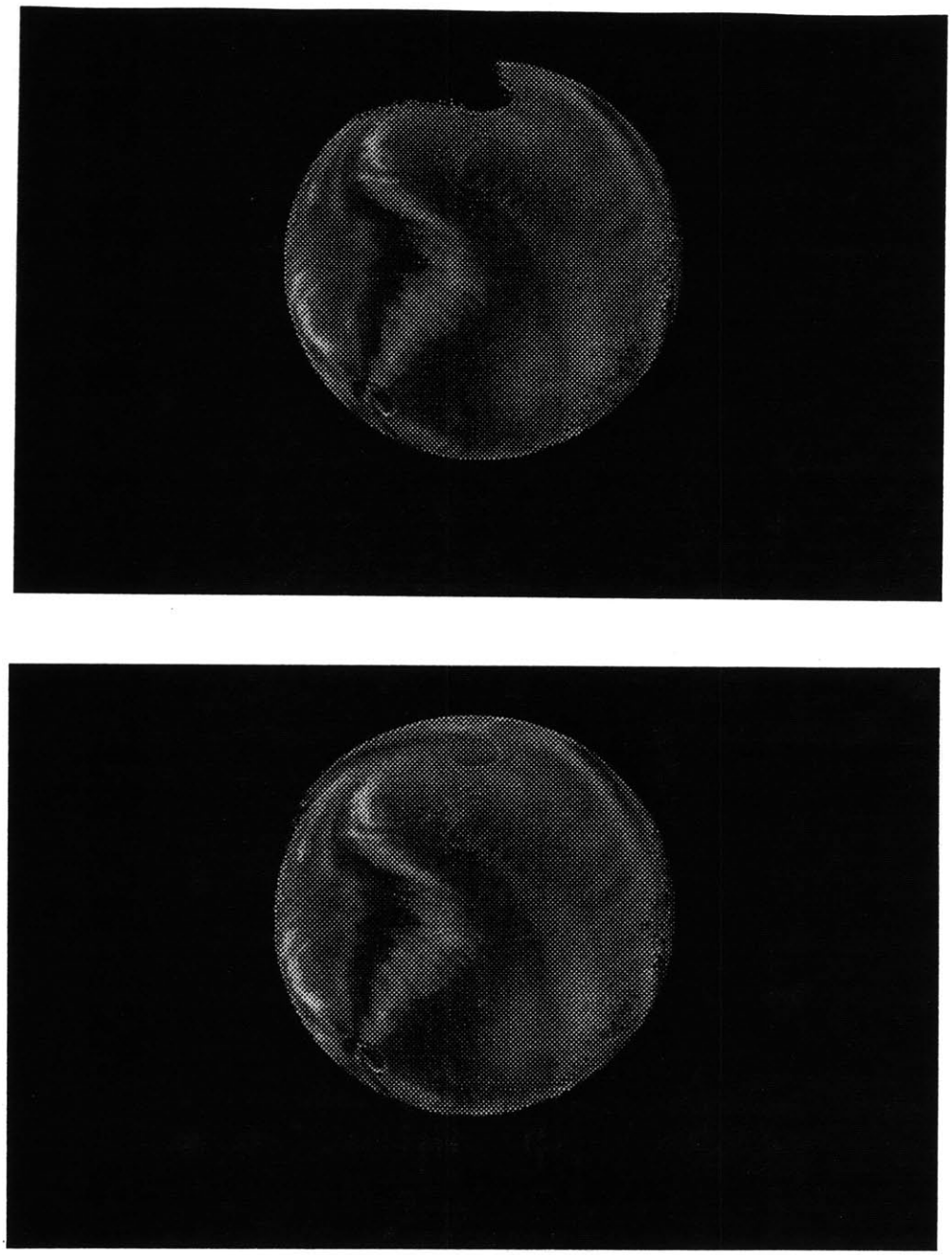

Figure 5-3: Continuation of the process in preceding figure.

Figure 5-4: Frames from input image sequence used to generate following figures.

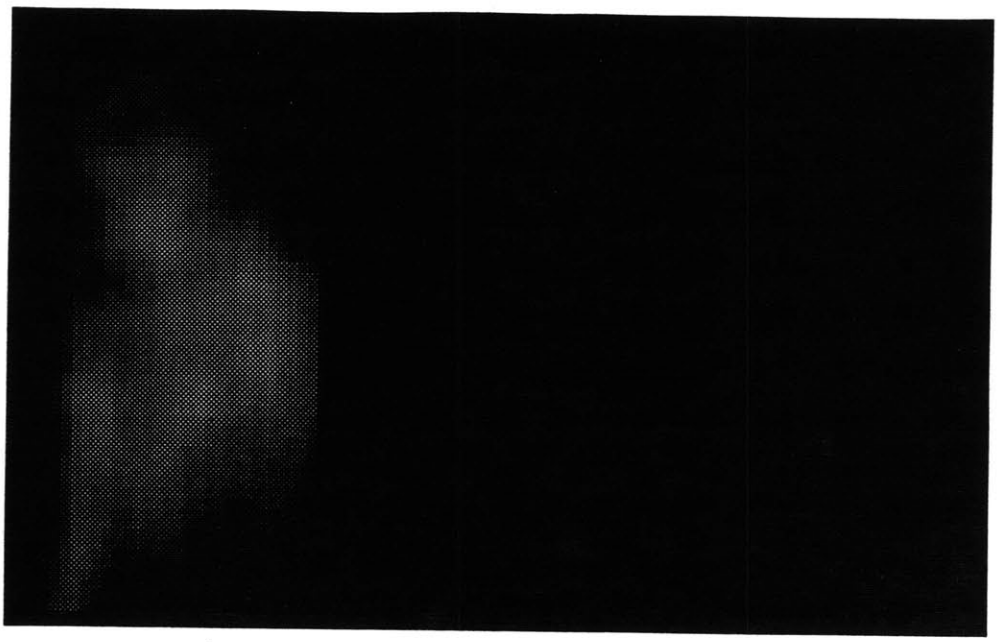

Figure 5-5: Low resolution depth-from-focus for one frame of input sequence.

in which every frame time's particles were used.

An animated videotape was produced consisting of the original sequence, the results of the depth-from-focus process (Figure 5-5), the range information from the motion estimator along with a graphical representation of the motion estimator's running estimate of the person's position and heading relative to the camera[6] (Figures 5-6 and 5-7), a variable-viewpoint rendering of the database built up over time (Figures 5-8 and 5-9), and the original movie reconstructed from the database and motion estimates, but viewed from a different viewpoint from that of the camera and with simulated lighting

[6]To produce this display, the system had to be told the figure's absolute position and heading at the time of the first frame.

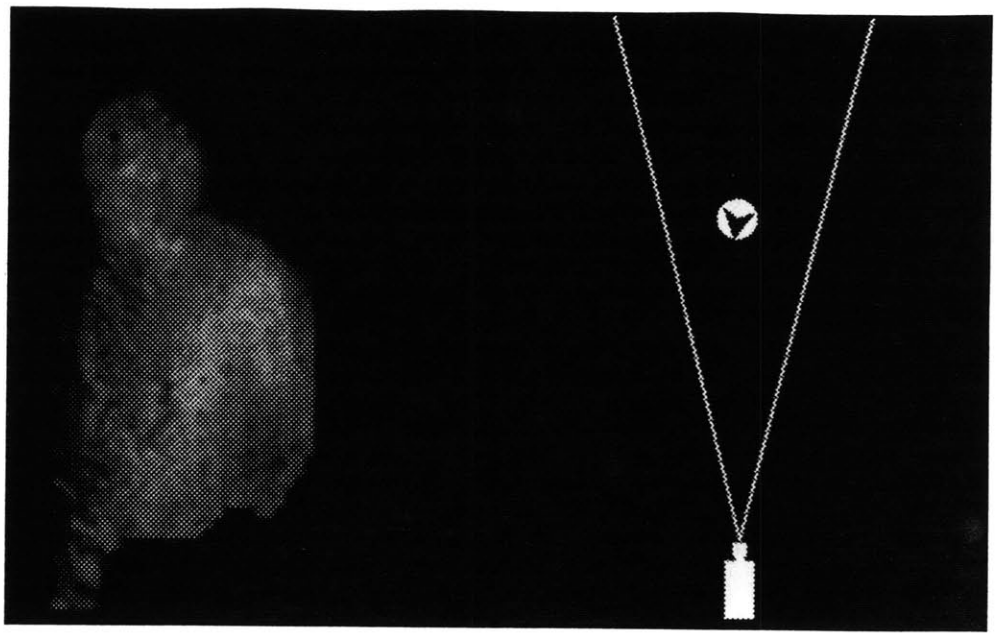

Figure 5-6: Range information after structure-from-motion (left), and estimate of figure's (x, z) position and heading relative to camera (right).

(Figure 5-10). In this reconstruction, the back wall may be seen to be complete.

The actual position and heading of the person at each frame time in the input movie were not measured, and thus the performance of the motion estimator must be evaluated visually. The graphical representation of the position and heading of the figure seems to track the input movie quite well; there is some error in the temporal vicinity of rapid changes of motion (such as the point at which the figure reverses direction) which is probably due to an overly wide motion smoothing filter. There appears to be an absolute heading error of roughly 15 degrees of y-axis rotation by the end of the sequence, though the database has been filled in long before this point. Ideally, the

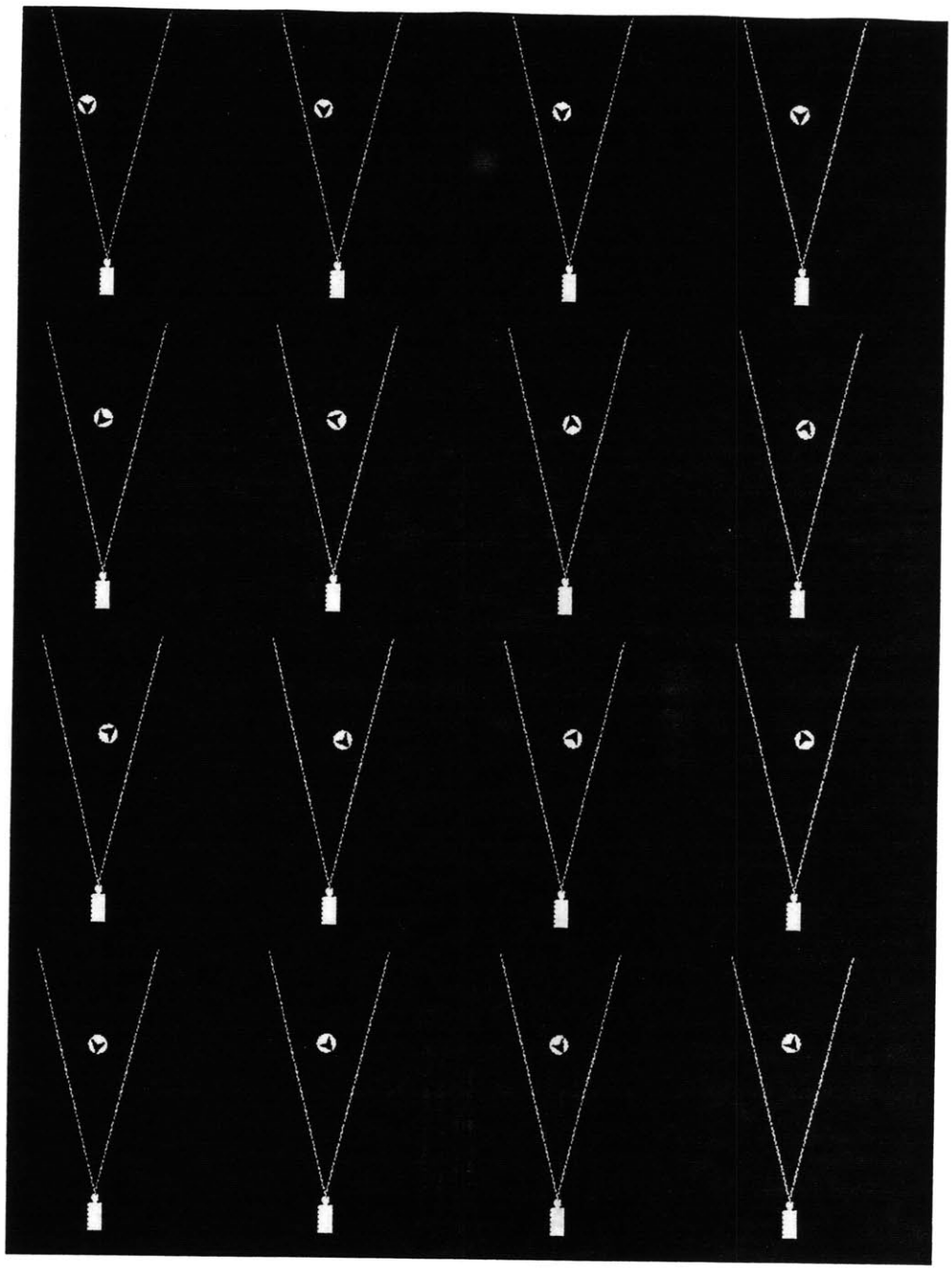

Figure 5-7: Frames showing motion estimator's estimate of position and heading of figure (compare with Figure 5-4).

Figure 5-8: Two views of the database built up from sequence shown in Figure 5-4.

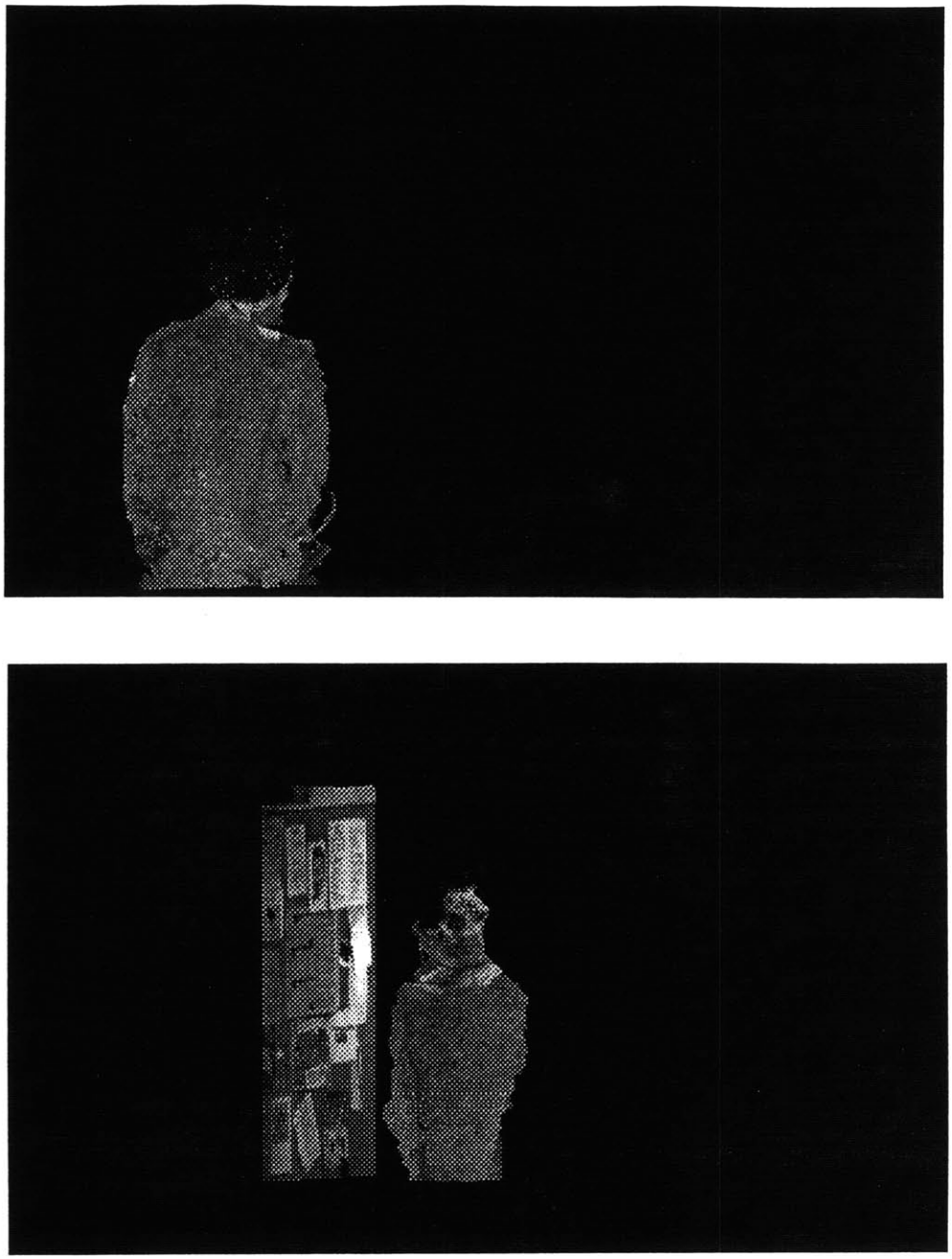

Figure 5-9: Two more views of database.

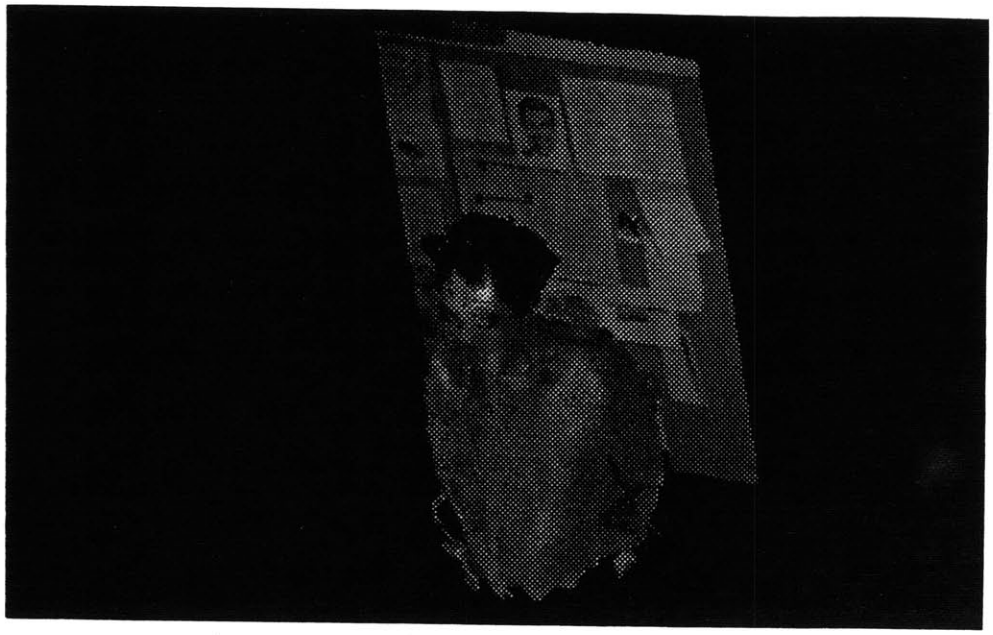

Figure 5-10: Frame of original movie reconstructed from a new viewpoint, and with simulated light source at upper left.

position and heading information should not be calculated in incremental form over such a long time period; absolute position and heading must be established (via some other technique) from time to time during an image sequence so that the incremental error does not build up to objectionable levels. Another way in which the system performance may be evaluated is to compare original frames with reconstructions from the database (Figures 5-11 and 5-12).

The database itself exhibits distortions and errors of three types: errors due to incorrect range information, errors due to incorrect estimation of absolute position of the figure, and errors arising from the building process. There are a few regions of the

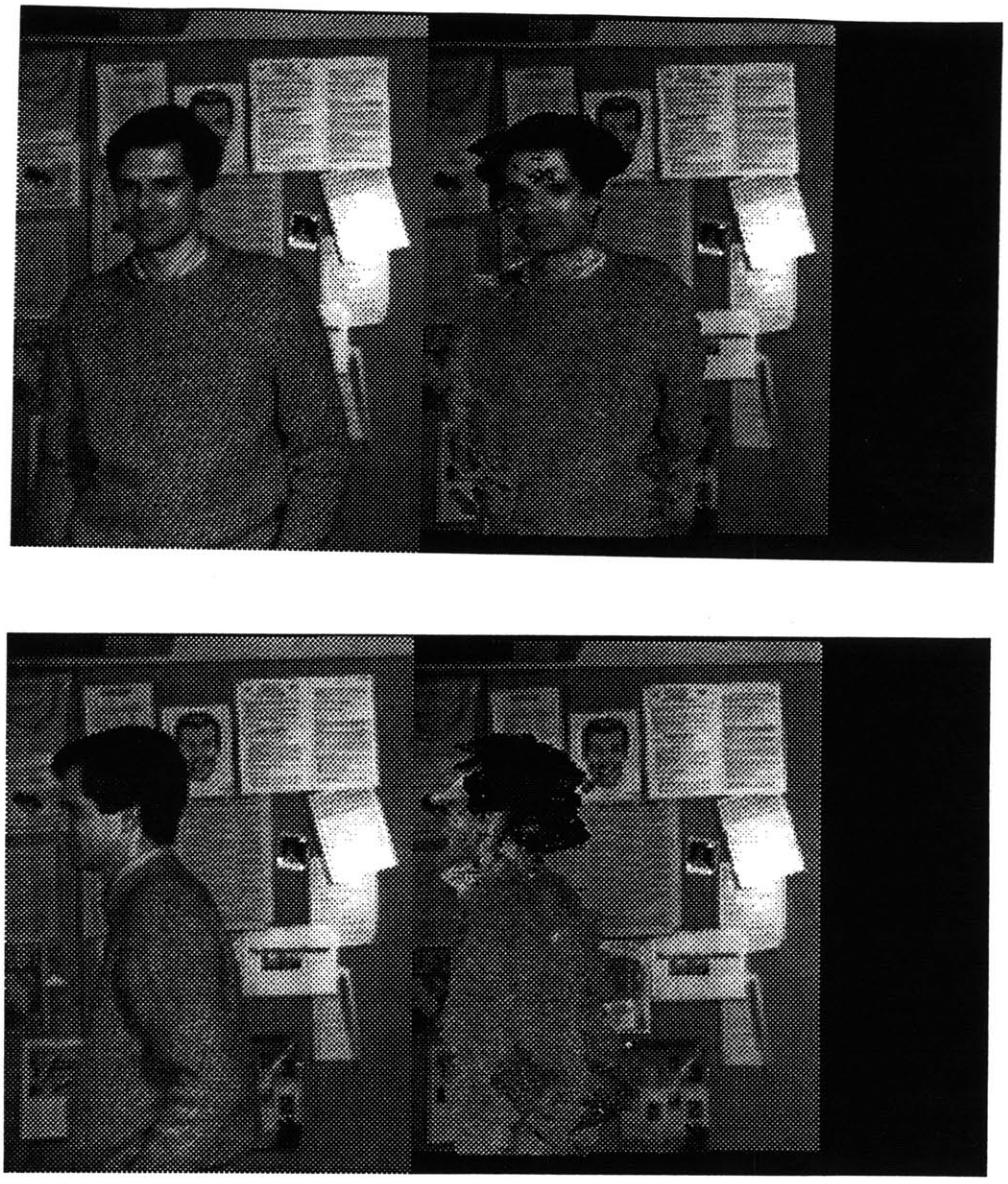

Figure 5-11: Two frames of original movie (left), and re-creations from the database.

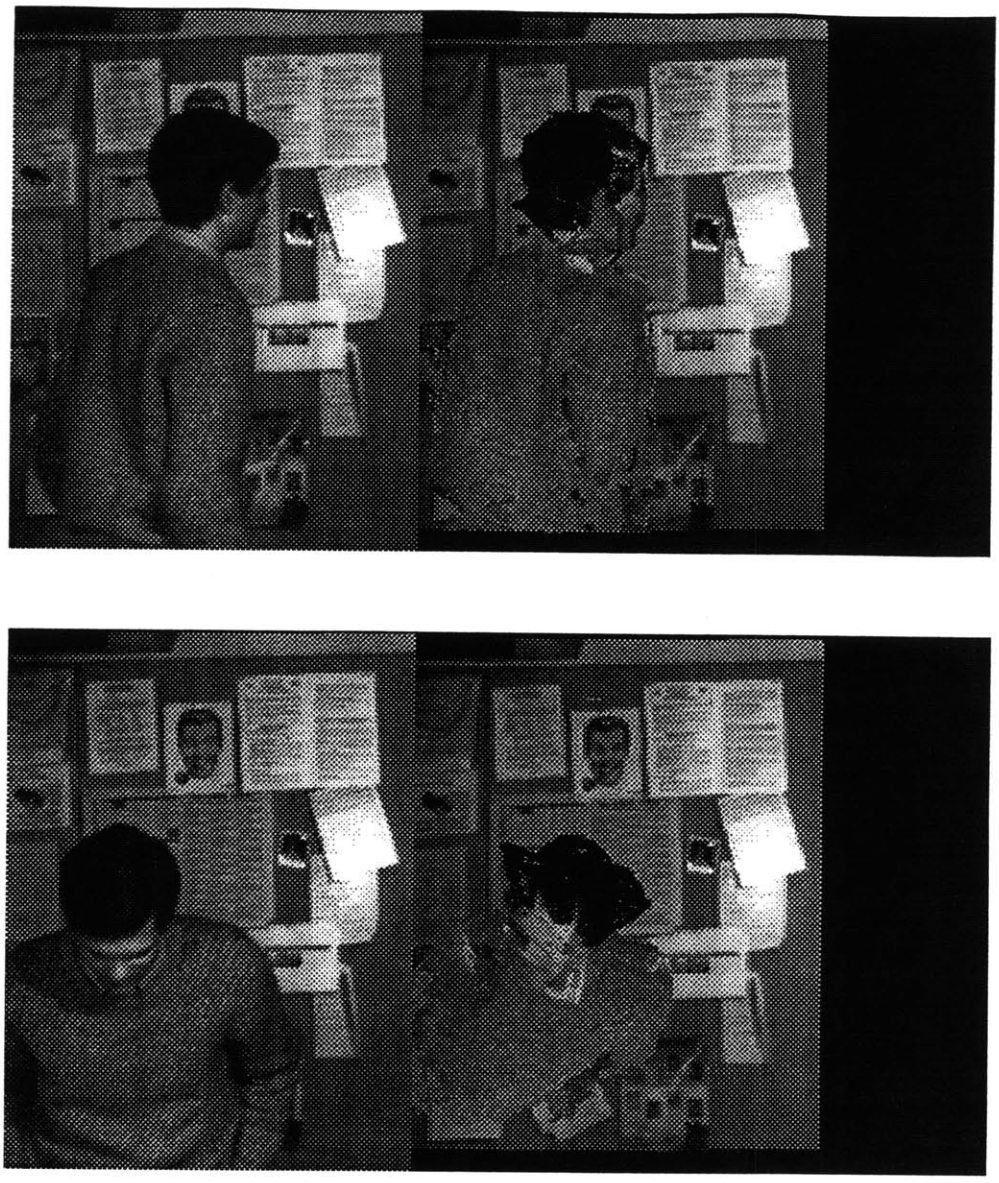

Figure 5-12: Frames of original movie (left), and re-creations from the database.

database that are slightly sparse, particularly on the figure's right side; this region did not face the camera for an appreciable time and during the period when it was near the line of sight the rotation of the figure was reversing direction (with the associated errors as discussed above).

Numerous material causes of error in both of these sequences may be identified. As observed in Chapter 4, the dual-aperture lens is not the most precise of optical instruments. Videotape noise reduces the accuracy of the depth-from-focus and motion estimation processes, while the lower bandwidth of the videotape (compared with digitizing video direct from the camera) prevents depth-from-focus from measuring very small degrees of defocusing. Additionally, the videotape recorder used was found to have slight scanner servo misadjustments – the video timing errors remaining after the time-base corrector were occasionally large enough to affect the motion estimates.

It would be unwise to attach too much importance to the data size figures or to make claims of coding efficiency based upon this limited experiment, but the following figures are given for the benefit of those who must see numerical results: for the "person" sequence, the input frames (long-depth-of-field side only) are 900 $\times$ 200 kilobytes, or 180,000 kilobytes, while the particle database came to just over 4000 kilobytes and the file of motion estimates in human-readable-and-editable form was approximately 30 kilobytes.

When considering the process described herein in signal coding terms rather than computer-graphics terms, it is important to realize that numerical comparison of a

reconstructed frame to the original may not yield a completely meaningful number. For example, a small error in position or orientation of an object might be completely unnoticeable to a viewer who does not have the original frame available, but would yield a large signal-to-noise ratio. An equal SNR might result from much worse (perhaps unacceptably so) database errors. A consideration of coding methods does suggest one interesting insight into the process, namely that of comparing the reconstruction with the original frame and intelligently using this "feedback" to improve the database.

5.5 Chapter Summary

A method has been proposed for employing range and motion information so as to combine views of objects and surroundings seen at different times in a movie in order to build a volumetric model of the scene contents. The resulting compact database has been used to reconstruct the original movie with computer graphics manipulations such as variable viewpoint and lighting.

Chapter 6

Conclusion

We'll not stop two moments, my dear Sir,—only, as we have got through these five volumes, (do, Sir, sit down upon a set—they are better than nothing) let us just look back upon the country we have passed through.—

Laurence Sterne

The Life and Opinions of Tristram Shandy

6.1 Summary and further investigations

The goal of this thesis project – a passive range- and motion-sensing camera capable of building a computer-graphics-like description of scenes before it – has been reached with a fair degree of success. To this end, several contributions to knowledge have been made. First, manipulations possible upon computer-graphics-like descriptions of real scenes have been demonstrated. A particular depth-from-focus algorithm which provides better results than have been reported elsewhere in the literature has been

developed and analyzed. It has been shown that the output of this process may be used to drive a spatiotemporal gradient motion estimator adapted to employ a parametric signal model, and the motions thus calculated used to solve for a new set of range constraints – an approach which provides greatly improved range data, and points toward the integration of still more sources of range information, as is done by biological vision. At the intersection of this range and motion information lies the demonstrated ability to build up models of objects by watching them as they move, as well as the compression of image sequences to a compact and modifiable set of object models and trajectories, where the receiver is not merely a display but takes on a computational character.

The author has learned a number of valuable lessons in the pursuit of this objective, but he hopes that he is not alone in having done so, and that the reader of this document has along with him gained an appreciation for the possibilities inherent in considering the camera as model-builder. The development of such a camera has, of course, merely begun and a great deal of research remains to be done in this exciting field. Some particularly important issues:

- Inasmuch as no one range-sensing modality performs well under all circumstances, more independent sources of range information should be added to the system in a cooperative manner. Methods that are already well-understood and could improve the overall accuracy of the system are shape-from-shading and stereopsis (whose computational requirements might be lessened markedly by using the

depth-from-focus information to guide correspondence searching).

- As Fourier transforms are already being taken of regions of every input frame, they might be employed in a transform-based motion estimator either in lieu of or in cooperation with the gradient-based motion estimator.

- Assumptions other than that of objects articulated of totally rigid parts are possible. The most general case is that of totally-deformable objects, but the computational and database complexity involved may preclude this model. A good compromise may be Ullman's incremental rigidity assumption [Ullman, 1983][Ullman, 1987], in which rigidity accounts for as much as possible of the motion of points on an object, with the remaining error interpreted as deformation. In any case, a means such as a hierarchical part tree is needed to express the articulation of multi-part objects, and constraints should be introduced so that the parts remain connected even given errors in motion estimation.

- *A priori* object knowledge, where a particular instance of an object seen by the camera is checked against an internal model – as in [Van Hove, 1987] – would improve scene segmentation and be nearly essential to part segmentation. It would also provide a check of the motion estimator by allowing the periodic estimation of *absolute* object orientation. The object knowledge possessed by the system might be specific, as in a Hollywood camera having a good model of a particular actor, or it might be generic, as in a camera which knows about the

ways in which human bodies are jointed and can apply this general knowldege to the specific person before it.

- Much investigation remains in the segmentation of dynamic scenes, including the use of segmentation by motion and the correct handling of objects which interocclude in complex ways.

- A parametric object description offers some advantages over a particle database, especially in compactness and rapid manipulation. In order to employ such a description, though, it will be necessary to develop an equally compact and efficient means of storing surface color information.

- An additional check on database and motion accuracy can be made by attempting to reconstruct the original frames from the database, and using the error to correct database and motion errors.

- Methods of employing data from multiple cameras over time should be explored.

- Finally, having developed a camera capable of building modifiable models of real scenes, the researcher is confronted with the challenge of coming up with applications and systems which will take advantage of its creative and communicative potential.

Tomorrow promises to be a busy day.

Bibliography

[Adelson, 1987] Adelson, E. H., E. Simoncelli, and R. Hingorani. "Orthogonal Pyramid Transforms for Image Coding." *SPIE Visual Communications and Image Processing II.* Cambridge MA, pp. 50-58, October 1987.

[Adiv, 1985] Adiv, G. "Determining Three-Dimensional Motion and Structure from Optical Flow Generated by Several Moving Objects." *IEEE Transactions on Pattern Analysis and Machine Intelligence.* PAMI-7:4, pp. 384-401, July 1985.

[Agin, 1973] Agin, G. J., and T. O. Binford. "Computer Description of Curved Objects." *Proceedings International Joint Conference on Artificial Intelligence.* Stanford University, pp. 629-640, August 1973.

[Archibald, 1986] Archibald, C., and M. Rioux. *Witness: A System for Object Recognition Using Range Images.* National Research Council of Canada Report No. 25588, January 1986.

[Ballard, 1983] Ballard, D. H., and O. A. Kimball. "Rigid Body Motion from Depth and Optical Flow." *Computer Vision, Graphics, and In-*

formation Processing. 22:1, pp. 95-115, April 1983.

[Bhanu, 1984] Bhanu, B. "Representation and Shape Matching of 3-D Objects." *IEEE Transactions on Pattern Analysis and Machine Intelligence.* PAMI-6, pp. 340-351, May 1984.

[Binford, 1971] Binford, T. O. "Visual Perception by Computer." *Proceedings of IEEE Conference on Systems and Control.* Miami FL, December 1971.

[Boissonat, 1981] Boissonat, J. D., and O. D. Faugeras. "Triangulation of 3-D Objects." *Proceedings of 7th International Joint Conference on Artificial Intelligence.* Vancouver Canada, pp. 658-660, August 1981.

[Born, 1970] Born, M., and E. Wolf. *Principles of Optics.* Pergamon Press, Oxford England, 1970.

[Boucher, 1968] Boucher, P. E. *Fundamentals of Photography.* Morgan and Morgan, New York NY, 1968.

[Bove, 1988] Bove Jr., V. M. "Pictorial Applications for Range Sensing Cameras." *SPIE Vol. 901 Proceedings of SPSE Electronic Imaging Devices and Systems '88.* Los Angeles CA, pp. 10-17, January 1988.

[Boyer, 1987] Boyer, K. L., and A. C. Kak. "Color-Encoded Structured Light for Rapid Active Ranging." *IEEE Transactions on Pattern Analysis and Machine Intelligence.* PAMI-9:1, pp. 14-28, January 1987.

[Broida, 1986] Broida, T., and R. Chellappa. "Estimation of Object Motion Parameters from Noisy Images." *IEEE Transactions on Pattern Analysis and Machine Intelligence.* PAMI-8:1, pp. 90-99, January 1986.

[Burt, 1983] Burt, P. J., and E. H. Adelson. "The Laplacian Pyramid as a Compact Image Code." *IEEE Transactions on Communications.* COM-31:4, pp. 532-540, April 1983.

[Chesnais, 1988] Chesnais, P. *Graphic/Photographic Simulation Systems.* S.M. thesis, Massachusetts Institute of Technology, Cambridge MA, January 1988.

[Constantinides, 1987] Constantinides, A. *Applied Numerical Methods with Personal Computers.* McGraw-Hill Book Company, New York NY, 1987.

[Cook, 1984] Cook, R. L., T. Porter, and L. Carpenter. "Distributed Ray Tracing." *Computer Graphics.* 18:3, pp. 137-145, July 1984.

[Crow, 1982] Crow, F. C. "A More Flexible Image Generation Environment." *Computer Graphics.* 16:3, pp. 9-18, July 1982.

[Darrell, 1988] Darrell, T., and K. Wohn. "Pyramid Based Depth from Focus." *Proceedings Computer Society Conference on Computer Vision and Information Processing.* Ann Arbor MI, pp. 504-509, June 1988.

[Foley, 1982] Foley, J. D., and A. van Dam. *Fundamentals of Interactive Computer Graphics.* Addison-Wesley, Reading MA, 1982.

[Fu, 1987] Fu, K. S., R. C. Gonzalez, and C. S. G. Lee. *Robotics: Control, Sensing, Vision, and Intelligence.* McGraw-Hill, New York NY, 1987.

[Gabor, 1946] Gabor, D. "Theory of Communication, Part III." *Journal of the IEE.* 93, pp. 429-457, 1946.

[Gardner, 1988] Gardner, W. A. *Statistical Spectral Analysis.* Prentice-Hall, Englewood Cliffs NJ, 1988.

[Garibotto, 1987] Garibotto, G., and P. Storace. "3-D Range Estimation from the Focus Sharpness of Edges." *Proceedings Fourth International Conference on Image Analysis and Processing.* Palermo Italy, pp. 321-328, September 1987.

[Gellert, 1975] Gellert, W., H. Kustner, M. Hellwich, and H. Kastner, eds. *The VNR Concise Encyclopedia of Mathematics.* Van Nostrand Reinhold, New York NY, 1975.

[Gennert, 1986a] Gennert, M. A. "Intensity-Based Stereo Matching." Massachusetts Institute of Technology Artificial Intelligence Laboratory, Cambridge MA, 1986.

[Gennert, 1986b] Gennert, M. A., and S. Negahdaripour. "Relaxing the Brightness Constancy Assumption in Optical Flow." Massachusetts Institute of Technology Artificial Intelligence Laboratory, Cambridge MA, 1986.

[Gerald, 1984] Gerald, C., and P. Wheatley. *Applied Numerical Analysis.* Addison-Wesley, Reading MA, 1984.

[EGibson, 1959] Gibson, E. J., J. J. Gibson, O. W. Smith, and H. Flock. "Motion Parallax as a Determinant of Perceived Depth." *Journal of Experimental Psychology.* 8:1, pp. 40-51, 1959.

[JGibson, 1950] Gibson, J. J. *The Perception of the Visual World.* Houghton Mifflin, Boston MA, 1950.

[Gil, 1983] Gil, B., A. Mitchi, and J. K. Aggarwal. "Experiments Combining Intensity and Range Edge Maps." *Computer Vision, Graphics, and Image Processing.* 21:3, pp. 395-411, September 1983.

[Girod, 1989] Girod, B. "Motion-Compensating Prediction with Fractional-Pel Accuracy." Manuscript submitted to *Proceedings of the IEEE.* January, 1989.

[Goldstein, 1950] Goldstein, H. *Classical Mechanics.* Addison-Wesley, Reading MA, 1950.

[Grimson, 1981] Grimson, W. E. L. *From Images to Surfaces.* MIT Press, Cambridge MA, 1981.

[Halfman, 1962] Halfman, R. L. *Dynamics: Particles, Rigid Bodies, and Systems.* Addison-Wesley, Reading MA, 1962.

[Hammond, 1981] Hammond, J. H. *The Camera Obscura: A Chronicle.* Adam Hilger Ltd., Bristol England, 1981.

[Harris, 1986] Harris, J. G. "The Coupled Depth/Slope Approach to Surface Reconstruction." Massachusetts Institute of Technology Arti-

ficial Intelligence Laboratory Technical Report No. 908, Cambridge MA, 1986.

[Haskell, 1974] Haskell, B. G. "Frame-to-Frame Coding of Television Pictures Using Two-Dimensional Fourier Transforms." *IEEE Transactions on Information Theory.* IT-20:1, pp. 119-120, January 1974.

[Hecht, 1974] Hecht, E., and A. Zajac. *Optics.* Addison-Wesley, Reading MA, 1974.

[Helmholtz, 1924] von Helmholtz, H. *Physiological Optics.* vol. 3, translated by J. P. Southall, Optical Society of America, 1924. Reprinted by Dover Publications, New York NY, 1962.

[Herriott, 1958] Herriott, D. R. "Recording Electronic Lens Bench." *Journal of the Optical Society of America.* 48:12, pp. 968-971, 1958.

[Hildreth, 1984] Hildreth, E. C. *The Measurement of Visual Motion.* MIT Press, Cambridge MA, 1984.

[Horn, 1968] Horn, B. K. P. "Focusing." Project MAC Artificial Intelligence Memo No. 160, Massachusetts Institute of Technology Artificial Intelligence Laboratory, Cambridge MA, 1968.

[Horn, 1981] Horn, B. K. P., and B. G. Schunck. "Determining Optical Flow." *Artificial Intelligence.* 17:1-3, pp. 185-203, August 1981.

[Horn, 1986] Horn, B. K. P. *Robot Vision.* MIT Press, Cambridge MA, 1986.

[Inokuchi, 1983] Inokuchi, S., T. Nita, F. Matsuda, and Y. Sakurai. "A Three Dimensional Edge-Region Operator for Range Pictures." *Proceedings of 6th International Joint Conference on Pattern Recognition.* Munich, Germany, pp. 918-920, October 1982.

[Jackins, 1980] Jackins, C. L., and S. L. Tanimoto. "Oct-Trees and Their Use in Representing Three-Dimensional Objects." *Computer Graphics and Image Processing.* 14, pp. 249-270, 1980.

[Jarvis, 1983] Jarvis, R. A. "A Perspective on Range Finding Techniques for Computer Vision." *IEEE Transactions on Pattern Analysis and Machine Intelligence.* PAMI-5:2, pp. 122-139, March 1983.

[Jenkins, 1968] Jenkins, G. M., and D. G. Watts. *Spectral Analysis and its Applications.* Holden-Day, San Francisco CA, 1968.

[Kappei, 1987] Kappei, F., and C. E. Liedtke. "Modelling of a Natural 3-D Scene Consisting of Moving Objects from a Sequence of Monocular TV Images." *Proceedings of SPIE Symposium on the Technologies for Optoelectronics.* Cannes, France, November 1987.

[Kass, 1987] Kass, M., A. Witkin, and D. Terzopoulos. "Snakes: Active Contour Models." *Proceedings First International Conference on Computer Vision.* London, England, pp. 259-268, June 1987.

[Kohn, 1987] Kohn, M. C. *Practical Numerical Methods: Algorithms and Programs.* Macmillan Publishing Company, New York NY, 1987

[Krause, 1987] Krause, E. A. *Motion Estimation for Frame-Rate Conversion.* Ph.D. thesis, Massachusetts Institute of Technology, Cambridge

MA, June 1987.

[Kuglin, 1975] Kuglin, C. D., and D. C. Hines. "The Phase Correlation Image Alignment Method." *Proceedings of the IEEE 1975 International Conference on Cybernetics and Society.* pp. 163-165, September 1975.

[Levine, 1973] Levine, M. D., D. A. O'Handley, and G. M. Yagi. "Computer Determination of Depth Maps." *Computer Graphics Image Processing.* Vol. 2, pp. 134-150, 1973.

[Linhardt, 1989] Linhardt, P. *Integration of Range Images from Multiple Viewpoints into a Particle Database.* S.M. thesis, Massachusetts Institute of Technology, Cambridge MA, February 1989.

[Lippman, 1987] Lippman, A., and W. Bender. "News and Movies in the 50 Megabit Living Room." *IEEE Globecom Proceedings.* Tokyo, November 1987.

[Lo, 1973] Lo, R. C., and J. A. Parikh. "A Study of the Application of Fourier Transforms to Cloud Movement Estimation from Satellite Photographs." University of Maryland Computer Science Center Technical Report TR-242, College Park MD, 1973.

[Mandelbrot, 1983] Mandelbrot, B. B. *The Fractal Geometry of Nature.* W. H. Freeman and Co., San Francisco CA, 1983.

[Markle, 1984] Markle, W. "The Development and Application of Colorization(R)." *SMPTE Journal.* 93:7, pp. 632-635, July 1984.

[Marr, 1982] Marr, D. *Vision.* W. H. Freeman and Company, New York NY, 1982.

[Martinez, 1986] Martinez, D. M. *Model-Based Motion Interpolation and its Application to Restoration and Interpolation of Motion Pictures.* Ph.D. thesis, Massachusetts Institute of Technology, Cambridge MA, August 1986.

[Mathias, 1985] Mathias, H., and R. Patterson. *Electronic Cinematography.* Wadsworth Publishing Company, Belmont CA, 1985.

[Matthies, 1987] Matthies, L., R. Szeliski, and T. Kanade. "Kalman Filter-Based Algorithms for Estimating Depth from Image Sequences." Carnegie-Mellon University Report CMU-CS-87-185, Pittsburgh PA, December 1987.

[Meagher, 1982] Meagher, D. "Geometric Modeling Using Octree Encoding." *Computer Graphics and Image Processing.* 19, pp. 129-147, 1982.

[Mohl, 1982] Mohl, R. F. *Cognitive Space in the Interactive Movie Map: An Investigation of Spatial Learning in Virtual Environments.* Ph.D. thesis, Massachusetts Institute of Technology, Cambridge MA, February 1982.

[Moravec, 1979] Moravec, H. P. "Visual Mapping by a Robot Rover." *Proceedings of the 6th International Joint Conference on Artificial Intelligence.* Tokyo, Japan, pp. 598-620, 1979.

[Mortenson, 1985] Mortenson, M. E. *Geometric Modeling.* Wiley, New York NY,

1985.

[Negahdaripour, 1987] Negahdaripour, S., and B. K. P. Horn. "Direct Passive Navigation." *IEEE Transactions on Pattern Analysis and Machine Intelligence.* PAMI-9:1, January 1987.

[Netravali, 1985] Netravali, A. N., and J. Salz. "Algorithms for Estimation of Three-Dimensional Motion." *AT&T Technical Journal.* 64:2, pp. 335-346, February 1985.

[Newman, 1979] Newman, W. F., and R. F. Sproull. *Principles of Interactive Computer Graphics.* McGraw-Hill, New York NY, 1979.

[Ninomiya, 1982] Ninomiya, Y., and Y. Ohtsuka. "A Motion-Compensated Interframe Coding Scheme for Television Pictures." *IEEE Transactions on Communications.* COM-30:1, pp. 201-211, January 1982.

[Nitzan, 1977] Nitzan, D., A. E. Brain, and R. O. Duda. "The Measurement and Use of Registered Reflectance and Range Data in Scene Analysis." *Proceedings of the IEEE.* 65:2, pp. 206-220, February 1977.

[Papoulis, 1977] Papoulis, A. *Signal Analysis.* McGraw-Hill, New York NY, 1977.

[Pearson, 1977] Pearson, J. J., D. C. Hines, Jr., S. Golosman, and C. D. Kuglin. "Video-Rate Image Correlation Processor." *SPIE Vol. 119 Applications of Digital Image Processing (IOCC 1977).* pp. 197-205, 1977.

[Pentland, 1982] Pentland, A. P. "Depth of Scene from Depth of Field." in *Proceedings DARPA Image Understanding Workshop.* Palo Alto CA, pp. 253-259, September 1982.

[Pentland, 1987a] Pentland, A. P. *The Parts of Perception.* Report No. CSLI-87-77, Center for the Study of Language and Information, Stanford CA, February 1987.

[Pentland, 1987b] Pentland, A. P. "A New Sense for Depth of Field." *IEEE Transactions on Pattern Analysis and Machine Intelligence.* PAMI-9:4, pp. 523-531, July 1987.

[Potmesil, 1981] Potmesil, M., and I. Chakravarty. "A Lens and Aperture Camera Model for Synthetic Image Generation." *Computer Graphics.* 15:3, pp. 297-305, August 1981.

[Pratt, 1978] Pratt, W. K. *Digital Image Processing.* John Wiley and Sons, New York NY, 1978.

[Prazdny, 1983] Prazdny, K. "On the Information in Optical Flows." *Computer Vision, Graphics, and Image Processing.* 22:2, pp. 239-259, May 1983.

[Press, 1986] Press, W. H., B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes: The Art of Scientific Computing.* Cambridge University Press, Cambridge England, 1986.

[Reeves, 1983] Reeves, W. T. "Particle Systems – A Technique for Modeling a Class of Fuzzy Objects." *Computer Graphics.* 17:3, pp. 359-376, July 1983.

[Reeves, 1985] Reeves, W. T., and R. Blau. "Approximate and Probabilistic Algorithms for Shading and Rendering Structured Particle Systems." *Computer Graphics.* 19:3, pp. 313-322, July 1985.

[Rives, 1986] Rives, P., E. Brueuil, and B. Espiau. "Recursive Estimation of 3D Features Using Optical Flow and Camera Motion." *Proceedings Conference on Intelligent Autonomous Systems.* Elsevier Science Publishers, pp. 522-532, December 1986.

[Rose, 1981] Rose, G. M. *Universal Focus Multiplanar Camera.* United States Patent 4,255,033, March 10, 1981.

[Ross, 1927] Ross, W. D. *The Works of Aristotle Translated into English, Volume VII - Problemata.* Clarendon Press, Oxford England, 1927.

[Schreiber, 1986] Schreiber, W. F. *Fundamentals of Electronic Imaging Systems.* Springer-Verlag, Berlin, 1986.

[ASmith, 1984] Smith, A. R. "Plants, Fractals, and Formal Languages." *Computer Graphics.* 18:3, pp. 1-10, July 1984.

[DSmith, 1985] Smith, D., and T. Kanade. "Autonomous Scene Description with Range Imagery." *Computer Vision, Graphics, and Image Processing.* 31:3, pp. 322-334, September 1985.

[Sobel, 1974] Sobel, I. "On Calibrating Computer Controlled Cameras for Perceiving 3-D Scenes." *Artificial Intelligence.* 5:2, pp. 185-198, 1974.

[Stokseth, 1969] Stokseth, P. A. "Properties of a Defocused Optical System." *Journal of the Optical Society of America.* 59:10, pp. 1314-1321, October 1969.

[Subbarao, 1988] Subbarao, M. "Parallel Depth Recovery by Changing Camera Features." *Proceedings IEEE 2nd International Conference on Computer Vision.* pp. 149-155, December 1988.

[Terzopoulos, 1987] Terzopoulos, D. "On Matching Deformable Models to Images." *Proceedings Optical Society of America Topical Meeting on Machine Vision.* Incline Village NV, pp. 160-167, March 1987.

[Terzopoulos, 1986] Terzopoulos, D. "Integrating Visual Information from Multiple Sources." in *From Pixels to Predicates: Recent Advances in Computational and Robotic Vision,* A. P. Pentland, ed. Ablex Publishing Corporation, Norwood NJ, 1986.

[Ullman, 1979] Ullman, S. *The Interpretation of Visual Motion.* MIT Press, Cambridge MA, 1979.

[Ullman, 1983] Ullman, S. "Maximizing Rigidity: The Incremental Recovery of 3D Structure from Rigid and Rubbery Motion." Artificial Intelligence Memo No. 721, Massachusetts Institute of Technology Artificial Intelligence Laboratory, Cambridge MA, June 1983.

[Ullman, 1987] Ullman, S., and A. Yuille. "Rigidity and Smoothness of Motion." Artificial Intelligence Memo No. 989, Massachusetts Institute of Technology Artificial Intelligence Laboratory, Cambridge MA, November 1987.

[Van Hove, 1987] Van Hove, P. "Model-Based Silhouette Recognition." *Proceedings of the IEEE Computer Society Workshop on Computer Vision.* Miami Beach FL, pp. 88-93, November-December 1987.

[Van Trees, 1968] Van Trees, H. L. *Detection, Estimation, and Modulation Theory Part I: Detection, Estimation, and Linear Modulation Theory.* John Wiley and Sons, New York NY, 1968.

[Webb, 1982] Webb, J. A., and J. K. Aggarwal. "Structure and Motion of Rigid and Jointed Objects." *Artificial Intelligence.* 19, pp. 107-130, 1982.

[Wheelock, 1977] Wheelock Jr., A. K. *Perspective, Optics, and Delft Artists Around 1650.* Garland Publishing Inc., New York NY, 1977.

[Whittaker, 1937] Whittaker, E. T. *A Treatise on the Analytical Dynamics of Particles and Rigid Bodies.* Cambridge University Press, Cambridge England, 1937.

[Yakimovsky, 1978] Yakimovsky, Y., and R. Cunningham. "A System for Extracting Three-Dimensional Measurements from a Stereo Pair of TV Cameras." *Computer Graphics Image Processing.* Vol. 7, pp. 195-210, 1978.

[Yelick, 1980] Yelick, S. *Anamorphic Image Processing.* S.B. thesis, Massachusetts Institute of Technology, Cambridge MA, May 1980.

[Zeltzer, 1985] Zeltzer, D. "Towards an Integrated View of 3-D Computer Animation." *The Visual Computer.* pp. 249-259, December 1985.

Acknowledgments

Sincere thanks to all who have helped in the carrying out of this project, particularly: Andy Lippman, who has advised me on this thesis and its two predecessors; my readers David Zeltzer and Alex Pentland for insightful comments; Nicholas Negroponte, for creating an environment in which this sort of research is not only possible but encouraged; Stephen Benton and Linda Peterson, for helping expedite the administrative aspects; Walter Bender, for all the nice colors and fonts, and for no fewer than six hundred fifty-two very good ideas, at last count; the Terminal Garden gang, particularly Pascal Chesnais, Pat Romano, John Watlington, and Bill Butera, for eliminating any possibility of boredom – or even normalcy; David Shane, for "Shanimating"; Paul Linhardt, who test-drove the particle database; Jean-Pierre Schott, for condensing about a thousand pages of information on spatiotemporal gradient motion estimators into a fifteen-minute explanation; Steve Fantone of Optikos Corporation, for technical assistance with the two-aperture lens; *The Tech*'s production department, for various graphical goodies; Gillian Galloway, for audiovisual supplies and refreshments; everyone who showed up for my final defense and suggested last-minute improvements; Mr. Pinkbunny, for enduring long hours under hot lights; Bernd Girod, for enthusiasm; Tamas and Susanne Sandor, for strategic advice and chocolate cake, respectively; my parents, who have helped me survive three of these things now; my wife Suzanne, for love, support, and not noticing thesis-induced aberrant behavior; and finally, all of you who read this and think about things a little differently as a result.

"THE MODEL-BUILDING CAMERA"