

Chapter 11

Depth

Calculating the distance of various points in the scene relative to the position of the camera is one of the important tasks for a computer vision system. A common method for extracting such depth information from intensity images is to acquire a pair of images using two cameras displaced from each other by a known distance. As an alternative, two or more images taken from a moving camera can also be used to compute depth information. In contrast to intensity images, images in which the value at each pixel is a function of the distance of the corresponding point in the scene from the sensor are called range images. Such images are acquired directly using range imaging systems. Two of the most commonly used principles for obtaining such range images are radar and triangulation. In addition to these methods in which the depth information is computed directly, 3-D information can also be estimated indirectly from 2-D intensity images using image cues such as shading and texture. These methods are described briefly in this chapter.

11.1 Stereo Imaging

The geometry of binocular stereo is shown in Figure 11.1. The simplest model is two identical cameras separated only in the x direction by a *baseline distance* b . The image planes are coplanar in this model. A feature in the scene is viewed by the two cameras at different positions in the image plane. The displacement between the locations of the two features in the image plane is called the *disparity*. The plane passing through the camera centers and

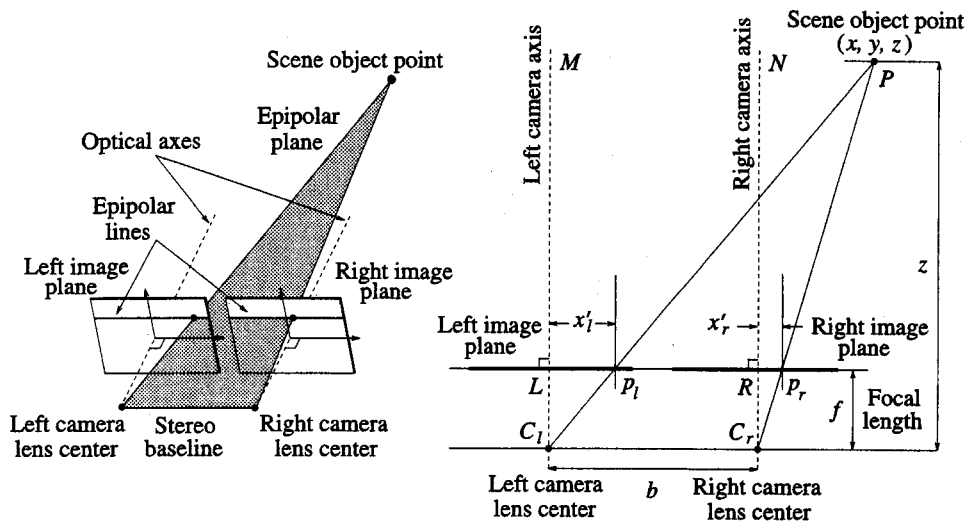


Figure 11.1: Any point in the scene that is visible in both cameras will be projected to a pair of image points in the two images, called a *conjugate pair*. The displacement between the positions of the two points is called the *disparity*.

the feature point in the scene is called the *epipolar plane*. The intersection of the epipolar plane with the image plane defines the *epipolar line*. For the model shown in the figure, every feature in one image will lie on the same row in the second image. In practice, there may be a vertical disparity due to misregistration of the epipolar lines. Many formulations of binocular stereo algorithms assume zero vertical disparity. Chapter 12 describes the relative orientation problem for calibrating stereo cameras.

Definition 11.1 A *conjugate pair* is two points in different images that are the projections of the same point in the scene.

Definition 11.2 *Disparity* is the distance between points of a conjugate pair when the two images are superimposed.

In Figure 11.1 the scene point P is observed at points p_l and p_r in the left and right image planes, respectively. Without loss of generality, let us

assume that the origin of the coordinate system coincides with the left lens center. Comparing the similar triangles PMC_l and p_lLC_l , we get

$$\frac{x}{z} = \frac{x'_l}{f} \quad (11.1)$$

Similarly, from the similar triangles PNC_r and p_rRC_r , we get

$$\frac{x - b}{z} = \frac{x'_r}{f} \quad (11.2)$$

Combining these two equations, we get

$$z = \frac{bf}{(x'_l - x'_r)} \quad (11.3)$$

Thus, the depth at various scene points may be recovered by knowing the disparities of corresponding image points.

Note that due to the discrete nature of the digital images, the disparity values are integers unless special algorithms are used to compute disparities to subpixel accuracy. Thus, for a given set of camera parameters, the accuracy of depth computation for a given scene point is enhanced by increasing the baseline distance b so that the corresponding disparity is large. Such wide-angle stereopsis methods introduce other problems, however. For instance, when the baseline distance is increased, the fraction of all scene points that are seen by both cameras decreases. Furthermore, even those regions that are seen by both cameras are likely to appear different in one image compared to the corresponding regions in the other image due to distortions introduced by perspective projection, making it difficult to identify conjugate pairs.

Before we discuss the problem of detecting and matching features in image pairs to facilitate stereopsis, we now briefly consider imaging systems in which the cameras are in any general position and orientation.

11.1.1 Cameras in Arbitrary Position and Orientation

Even when the two cameras are in any general position and orientation, the image points corresponding to a scene point lie along the lines of intersection between the image planes and the epipolar plane containing the scene point and the two lens centers as shown in Figure 11.2. It is clear from this figure that the epipolar lines are no longer required to correspond to image rows.

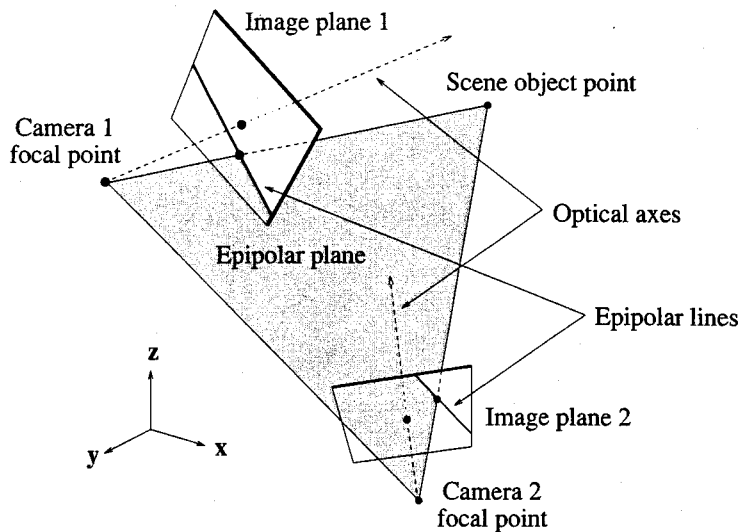


Figure 11.2: Two cameras in arbitrary position and orientation. The image points corresponding to a scene point must still lie on the epipolar lines.

In certain systems, the cameras are oriented such that their optical axes intersect at a point in space. In this case, the disparity is relative to the vergence angle. For any angle there is a surface in space corresponding to zero disparity as shown in Figure 11.3. Objects that are farther than this surface have disparity greater than zero, and objects that are closer have disparity less than zero. Within a region the disparities are grouped into three pools:

$$\begin{array}{rcl}
 + & d > 0 \\
 - & d < 0 \\
 0 & d = 0
 \end{array}$$

These pools are then used to resolve ambiguous matches.

More recent research work has addressed the issue of dynamically controlling the position, orientation, and other camera parameters to facilitate better image analysis. In systems known as *active vision systems*, the image analysis process dynamically controls camera parameters and movements. Computing the depths of various points in a scene is a common task in such systems.

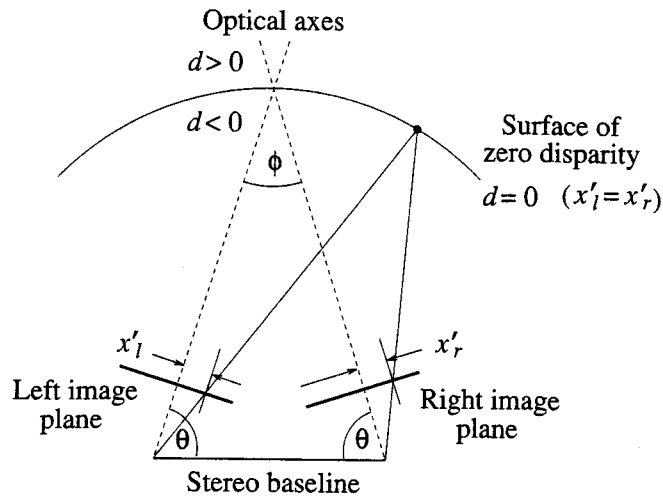


Figure 11.3: Stereo cameras focused at a point in space. The angle of the cameras defines a surface in space for which the disparity is zero.

11.2 Stereo Matching

Implicit in the stereopsis technique is the assumption that we can identify conjugate pairs in the stereo images. Detecting conjugate pairs in stereo images, however, has been an extremely challenging research problem known as the *correspondence problem*.

The correspondence problem can be stated as follows: *for each point in the left image, find the corresponding point in the right image*. To determine that two points, one in each image, form a conjugate pair, it is necessary to measure the similarity of the points. Clearly, the point to be matched should be distinctly different from its surrounding pixels; otherwise (e.g., for pixels in homogeneous intensity regions), every point would be a good match. Thus, before stereo matching, it is necessary to locate matchable features. Both edge features and region features have been used in stereo matching.

The implication of selecting a subset of all image pixels for matching is that depth is only computed at these feature points. Depth at other points is then estimated using interpolation techniques.

Note that the epipolar constraint significantly limits the search space for finding conjugate pairs. However, due to measurement errors and other uncertainties in camera position and orientation, matching points may not

occur exactly on the estimated epipolar lines in the image plane; in this case, a search in a small neighborhood is necessary.

11.2.1 Edge Matching

We first present an algorithm for binocular stereo. The basic idea behind this and similar algorithms is that features are derived from the left and right images by filtering the images, and the features are matched along the epipolar lines. In this discussion, the epipolar lines are along the image rows. This algorithm uses edges detected by the first derivative of Gaussian. Edges computed using the gradient of Gaussian are more stable with respect to noise. The steps in the stereo algorithm are:

1. Filter each image in the stereo pair with Gaussian filters at four filter widths such that each filter is twice as wide as the next smaller filter. This can be done efficiently by repeated convolution with the smallest filter.
2. Compute the edge positions within the row.
3. Match edges at coarse resolutions by comparing their orientations and strengths; clearly, horizontal edges cannot be matched.
4. Refine the disparity estimates by matching at finer scales.

Note that computing the edge pixels to subpixel resolution would improve the precision of depth computation. In order to simplify the matching processes, the search for a match for each feature in one image takes place along the corresponding epipolar line in the second image for a limited distance centered around its expected position. In addition, the orientation of edges is recorded in 30° increments and the coarse orientations are used in matching. The orientation can be efficiently computed by coarsely quantizing the x and y partial derivatives and using a lookup table. One could also evaluate potential matches by using a composite norm that includes terms that penalize differences in edge orientation, edge contrast, and other measures of lack of similarity between potential features.

With active convergent cameras, the edges are matched at a coarse scale, and then the angle between the cameras is adjusted so that the region has

a disparity of around zero, and the matching is redone at a finer scale. This adjustment limits the value of the maximum disparity and hence reduces the number of false matches and speeds up the matching process even when a small-scale filter is used to compute the disparity accurately. The matching process must begin at a coarse scale to determine an approximate value for the disparity. There are fewer edges at a coarse filter scale, so there is little chance of false matches.

11.2.2 Region Correlation

An important limitation of edge-based methods for stereo matching is that the value of the computed depth is not meaningful along occluding edges where the depth is not well defined. Along these edges the value of the depth is anywhere from the distance of the foreground object's occluding edge to the distance of the background scene point. In particular, for curved objects occluding edges are silhouette edges, and the observed image plane curves in the two images do not correspond to the same physical edge. Unfortunately, strong edges in the image plane are detected only along such occluding edges unless the object has other high-contrast nonoccluding edges or other features. Thus one of the primary problems in recovering depth is the problem of identifying more features distributed throughout the image as candidates for correspondence. One of the many methods developed for finding potential features for correspondence is to identify *interesting points* in both images and match these points in the two images using *region correlation* methods.

Detection of Interesting Points in Regions

In matching points from two images, we need points that can be easily identified and matched in the images. Obviously, the points in a uniform region are not good candidates for matching. The interest operator finds areas of image with high variance. It is expected that there will be enough of such isolated areas in images for matching.

The variances along different directions computed using all pixels in a window centered about a point are good measures of the distinctness of the point along different directions. The directional variances are given by

$$I_1 = \sum_{(x,y) \in S} [f(x,y) - f(x,y+1)]^2 \quad (11.4)$$

$$I_2 = \sum_{(x,y) \in S} [f(x,y) - f(x+1,y)]^2 \quad (11.5)$$

$$I_3 = \sum_{(x,y) \in S} [f(x,y) - f(x+1,y+1)]^2 \quad (11.6)$$

$$I_4 = \sum_{(x,y) \in S} [f(x,y) - f(x+1,y-1)]^2, \quad (11.7)$$

where S represents the pixels in the window. Typical window sizes range from 5×5 to 11×11 pixels. Since simple edge points have no variance in the direction of the edge, the minimum value of the above directional variances is taken as the *interest* value at the central pixel, (x_c, y_c) . This eliminates edge pixels from consideration since an edge pixel in one image would match all pixels along the same edge in the second image, making it difficult to determine exact disparity (especially when the edge is along the epipolar line). Thus, we have

$$I(x_c, y_c) = \min(I_1, I_2, I_3, I_4). \quad (11.8)$$

Finally, to prevent multiple neighboring points from being selected as *interesting* for the same feature, feature points are chosen where the interest measure has a local maxima. A point is considered a “good” interesting point if, in addition, this local maxima is greater than a preset threshold.

Once features are identified in both images, they can be matched using a number of different methods. A simple technique is to compute the correlation between a small window of pixels centered around a feature in the first image and a similar window centered around every potential matching feature in the second image. The feature with the highest correlation is considered as the match. Clearly, only those features which satisfy the epipolar constraint are considered for matching. To allow for some vertical disparity, features which are near the epipolar line are also included in the potential matching feature set.

Consider two images f_1 and f_2 . Let the pair of candidate feature points to be matched have a disparity of (d_x, d_y) . Then a measure of similarity between the two regions centered around the features is given by the *correlation*

coefficient $r(d_x, d_y)$ defined as

$$r(d_x, d_y) = \frac{\sum_{(x,y) \in S} [f_1(x, y) - \bar{f}_1] [f_2(x + d_x, y + d_y) - \bar{f}_2]}{\left\{ \sum_{(x,y) \in S} [f_1(x, y) - \bar{f}_1]^2 \sum_{(x,y) \in S} [f_2(x + d_x, y + d_y) - \bar{f}_2]^2 \right\}^{1/2}}. \quad (11.9)$$

Here \bar{f}_1 and \bar{f}_2 are the average intensities of the pixels in the two regions being compared, and the summations are carried out over all pixels within small windows centered around the feature points.

Instead of using the image intensity values in the above equation, the accuracy of correlation is improved by using thresholded signed gradient magnitudes at each pixel. This is done by computing the gradient at each pixel in the two images without smoothing and then mapping these into three values, -1 , 0 , and 1 , using two thresholds, one above zero and the other below zero. This transforms the images into square waves that produce more sensitive correlations. If this is done, it is not necessary to include the normalization terms shown in the equation for correlation, and $r(d_x, d_y)$ simplifies to the sum of the products of corresponding pixel values.

In most situations, the depths of scene points corresponding to nearby feature points are likely to be close to one another. This heuristic is exploited in the iterative relaxation method described in Section 14.3.

As noted earlier, stereo matching based on features produces a sparse depth map at scene points which correspond to the image feature points. Surface interpolation or approximation must be performed on the sparse depth map to reconstruct the surface as described in Chapter 13.

One of the principal difficulties in stereo reconstruction is in the selection of *interesting points*. Such points are typically selected based on high local variance in intensity. Unfortunately, such points occur more frequently at corners and other surface discontinuities where the smoothness constraint does not hold. In some machine vision applications, this problem is solved by using structured light. Patterns of light are projected onto the surface, creating interesting points even in regions which would be otherwise smooth as shown in Figure 11.4.

Finding and matching such points are further simplified by knowing the geometry of the projected patterns. Since these patterns create artificial texture on the surfaces, the *shape from texture* techniques described in Chapter 7 may also be used.



Figure 11.4: Patterns of light are projected onto a surface to create interesting points on an otherwise smooth surface.

11.3 Shape from X

In addition to the stereo imaging method described above, numerous other methods known as *shape from X* techniques have been developed for extracting shape information from intensity images. Many of these methods estimate local surface orientation rather than absolute depth at each point. If the actual depth to at least one point on each object is known, then the depth at other points on the same object can be computed by integrating the local surface orientation. Hence these methods are called indirect methods for depth computation. We briefly describe some of these methods here and provide pointers to other chapters where they are described in more detail.

Photometric Stereo

In the photometric stereo method, three images of the same scene are obtained using light sources from three different directions. Both camera and objects in the scene are required to be stationary during the acquisition of the three images. By knowing the surface reflectance properties of the objects in the scene, the local surface orientation at points illuminated by all

three light sources can be computed. This method is described in detail in Chapter 9. One of the important advantages of the photometric stereo method is that the points in all three images are perfectly registered with one another since both camera and scene are stationary. Thus, this method does not suffer from the correspondence problem. The primary disadvantages of this method are that it is an indirect method and it may not be practical to employ an imaging system in which the illumination is so carefully controlled.

Shape from Shading

Shape from shading methods exploit the changes in the image intensity (shading) to recover surface shape information. This is done by calculating the orientation of the scene surface corresponding to each point (x', y') in the image. In addition to the constraint imposed by the radiometric principles, shape from shading methods assume that the surfaces are smooth in order to calculate surface orientation parameters. This method is described in detail in Chapter 9. Clearly, shape from shading is an indirect method for depth computation. Furthermore, the smoothness constraint is not satisfied at all points and the surface reflectance properties are not always known accurately, resulting in inaccurate reconstructions.

Shape from Texture

Image plane variations in the texture properties such as density, size, and orientation are the cues exploited by shape from texture algorithms. For example, the *texture gradient*, defined as the magnitude and direction of maximum change in the primitive size of the texture elements, determines the orientation of the surface. Quantifying the changes in the shape of texture elements (e.g., circles appearing as ellipses) is also useful to determine surface orientation. From images of surfaces with textures made up of regular grids of lines, possibly due to structured lighting (described in the following section), orientation may be uniquely determined by finding the *vanishing points*. Besides being indirect methods for depth computation, shape from texture methods also suffer from difficulties in accurately locating and quantifying texture primitives and their properties. Shape from texture techniques are described in Chapter 7.

Shape from Focus

Due to the finite depth of field of optical systems (see Chapter 8), only objects which are at a proper distance appear focused in the image whereas those at other depths are blurred in proportion to their distances. Algorithms to exploit this blurring effect have been developed. The image is modeled as a convolution of focused images with a point spread function determined by the camera parameters and the distance of the object from the camera. The depth is recovered by estimating the amount of blur in the image and using the known or estimated line spread function. Such reconstruction problems are mathematically ill posed. However, in some applications, especially those requiring qualitative depth information, depth from focus methods are useful.

Shape from Motion

When images of a stationary scene are acquired using a moving camera, the displacement of the image plane coordinate of a scene point from one frame to another depends on the distance of the scene point from the camera. This is thus similar to the stereo imaging described in earlier sections. Alternatively, a moving object also produces motion disparity in image sequences captured by a stationary camera. Such a disparity also depends upon the position and velocity of the object point. Methods for recovering structure and motion of objects are described in detail in Chapter 14.

11.4 Range Imaging

Cameras which measure the distance to every scene point within the viewing angle and record it as a two-dimensional function are called *range imaging systems*, and the resulting images are called *range images*. Range images are also known as *depth maps*. A range image is shown in Figure 11.5.

Two of the most commonly used principles for range imaging are triangulation and radar. *Structured lighting* systems, which are used extensively in machine vision, make use of the principle of triangulation to compute depth. *Imaging radar* systems use either acoustic or laser range finders to compute the depth map.

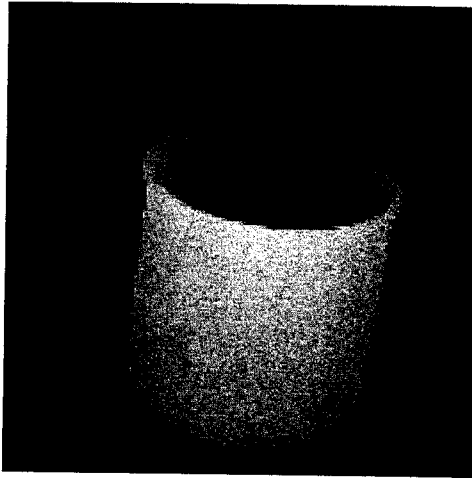


Figure 11.5: A range image of a coffee mug.

11.4.1 Structured Lighting

Imaging using structured lighting refers to systems in which the scene is illuminated by a known geometrical pattern of light. In a simple point projection system, a light projector and a camera are separated by a baseline distance b as shown in Figure 11.6. The object coordinates (x, y, z) are related to the measured image coordinates (x', y') and projection angle θ by

$$[x \ y \ z] = \frac{b}{f \cot \theta - x'} [x' \ y' \ f]. \quad (11.10)$$

The range resolution of such a triangulation system is determined by the accuracy with which the angle θ and the horizontal position of the image point x' can be measured.

To compute the depth at all points, the scene is illuminated one point at a time in a two-dimensional grid pattern. The depth at each point is then calculated using the above equation to obtain a two-dimensional range image. Because of its sequential nature, this method is slow and is not suitable for use with dynamically changing scenes. In a typical structured lighting system either planes of light or two-dimensional patterns of light are projected on the scene. The projected pattern of light on the surfaces of the objects in the scene is imaged by a camera which is spatially displaced with respect to the source of illumination. The observed image of the light patterns contain

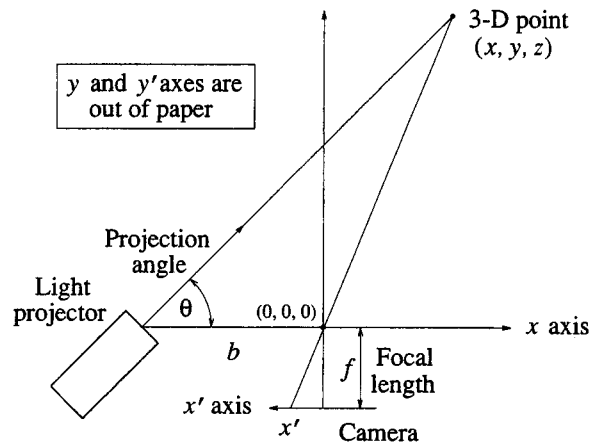


Figure 11.6: Camera-centered triangulation geometry [26].

distortions as determined by the shape and orientation of object surfaces on which the patterns are projected. This is illustrated in Figure 11.7 (also see Figure 11.4). Note that the light pattern as seen by the camera contains discontinuities and changes in orientation and curvature. The 3-D object coordinate corresponding to any point in the image plane may be calculated by computing the intersection of the camera's line of sight with the light plane. To obtain the complete object description, either the light source is panned as shown in the figure or the object is moved on a conveyor belt to acquire multiple images. Different surfaces in the object are detected by clustering stripes of light having similar spatial attributes.

In dynamically changing situations it may not be practical to project light stripes in sequence to acquire the complete image set covering the entire scene. Note that if multiple stripes of light are simultaneously projected to acquire the entire depth map, there would be potential ambiguities in matching stripe segments resulting from object surfaces at different depths. In such a case, patterns of multiple stripes of light in which each stripe is uniquely encoded are projected. For example, using a binary encoding scheme, it is possible to acquire a complete set of data by projecting only $\log_2 N$ patterns where $(N - 1)$ is the total number of stripes. This method is illustrated in Figure 11.8 for $N = 8$.

Each of the seven stripes has a unique binary code from (001) to (111).

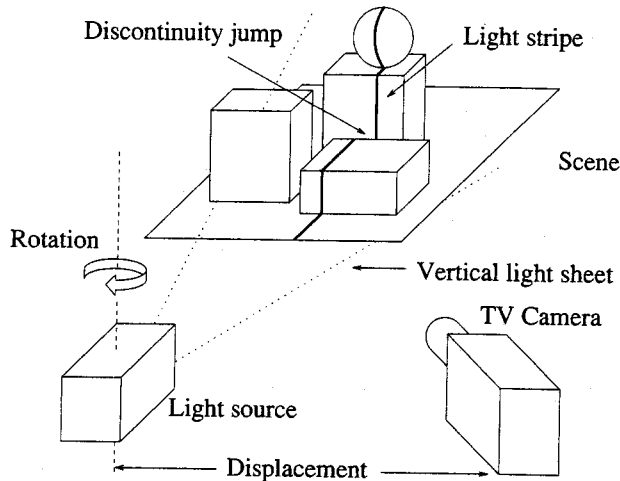


Figure 11.7: Illustration of striped lighting technique [131].

Since $\log_2 8$ is 3, only three images are acquired. Each image is identified by the bit position 1, 2, or 3 of the 3-bit binary code. A particular stripe of light is turned ON in an image only if its corresponding bit position is 1. For example, stripe 2 (010) is ON only in the second image, whereas stripe 7 (111) is ON in all three images. The stripes in the set of three images are now uniquely identified and hence there would be no ambiguities in matching stripe segments. In rapidly changing scenes, a single color-coded image is used instead of several binary-encoded images.

Structured lighting techniques have been used extensively in industrial vision applications in which it is possible to easily control the illumination of the scene. In a typical application, objects on a conveyor belt pass through a plane of light creating a distortion in the image of the light stripe. The profile of the object at the plane of the light beam is then calculated. This process is repeated at regular intervals to recover the shape of the object.

The primary drawback of structured lighting systems is that it is not possible to obtain data for object points which are not visible to either the light source or the imaging camera.

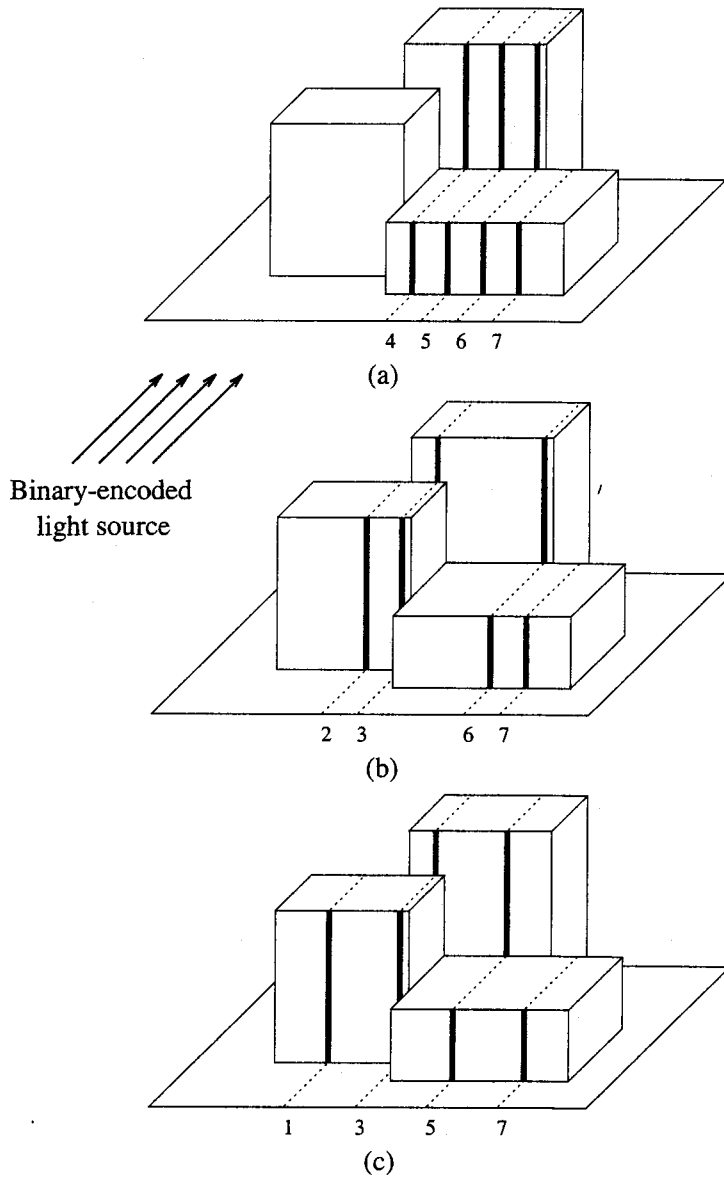


Figure 11.8: Illustration of binary-coded structured lighting where the sequence of projections determines the binary code for each stripe.

11.4.2 Imaging Radar

A second method for range imaging is *imaging radar*. In a time-of-flight pulsed radar, the distance to the object is computed by observing the time difference between the transmitted and received electromagnetic pulse. The depth information can also be obtained by detecting the phase difference between the transmitted and received waves of an amplitude-modulated beam or the beat frequency in a coherently mixed transmitted and received signal in a frequency-modulated beam. Several commercial laser beam imaging systems are built using these principles.

Range images are useful due to their explicit specification of depth values. At one time it was believed that if depth information is explicitly available, later processing would be easy. It became clear that though the depth information helps, the basic task of image interpretation retains all its difficulties.

11.5 Active Vision

Most computer vision systems rely on data captured by systems with fixed characteristics. These include both passive sensing systems such as video cameras and active sensing systems such as laser range finders. In contrast to these modes of data capture, it is argued that an active vision system in which the parameters and characteristics of data capture are dynamically controlled by the scene interpretation system is crucial for perception. The concept of active vision is not new. Biological systems routinely acquire data in an active fashion. Active vision systems may employ either passive or active sensors. However, in an active vision system, the state parameters of the sensors such as focus, aperture, vergence, and illumination are controlled to acquire data that would facilitate the scene interpretation task. Active vision is essentially an intelligent data acquisition process controlled by the measured and calculated parameters and errors from the scene. Precise definitions of these scene- and context-dependent parameters require a thorough understanding of not only the properties of the imaging and processing system, but also their interdependence. Active vision is a very active area of research.

Further Reading

The Marr-Poggio-Grimson algorithm for binocular stereo is described in the book on stereo by Grimson [92] and in the book on computational vision by Marr [163]. Barnard and Fischler [20] published a survey paper on binocular stereo. Stereo vision is also described in detail in the book by Faugeras [78]. An iterative algorithm for disparity analysis of images is described by Barnard and Thompson [21]. Many different forms of stereo have been investigated, including trinocular stereo [12, 98] and motion stereo [127, 185]. For solving difficulties in correspondence problems, different features such as interesting points [172], edge and line segments [11, 16, 167, 255], regions [60, 161, 167], and multiple primitives [162] have been tried to make stereo practical and useful.

One of the most active research topics in computer vision has been shape from X techniques. Shape from shading methods are described in the book with the same title [112]. The problem of concave objects resulting in secondary illuminations has been studied by Nayar et al. [179]. A recent approach to shape from focus is given in [180]. Shape from texture methods have been investigated by several groups [5, 36, 136].

Range image sensing, processing, interpretation, and applications are described in detail in the book [128]. Various methods for acquiring range images and a relative comparison of their merits are given by Besl [26]. An earlier survey by Jarvis includes not only direct range measuring techniques but also methods in which the range is calculated from 2-D image cues [131]. Boyer and Kak [44] describe a method in which color coding is used to acquire the range information using a single image. The classical paper by Will and Pennington [250] describes grid coding and Fourier domain processing techniques to locate different planar surfaces in the scene.

Arguments in favor of active perception systems and a control strategy for an active vision system were presented by Bajcsy [14]. Krotkov [150] has described a stereo image capture system in which the focus, zoom, aperture, vergence, and illumination are actively controlled to obtain depth maps. Aloimonos et al. [8] have described the advantages of active vision systems for performing many computer vision tasks such as shape recovery from image cues. Ahuja and Abbot [3] have integrated disparity, camera vergence, and lens focus for surface estimation in an active stereo system.

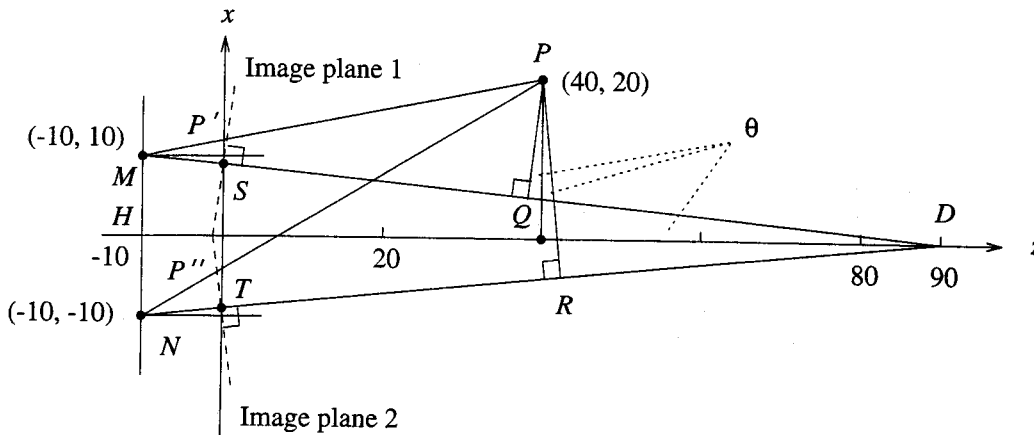


Figure 11.9: Convergent binocular imaging system for Exercise 11.3.

Exercises

- 11.1 A cube with 10 cm long edges is being imaged using a range camera system. The camera axis passes through two diagonally opposite vertices of the cube, and the nearest vertex along the axis is 10 cm away from the camera center. The intensity recorded by the camera is equal to $1000/d$ where d is the distance from the camera plane measured along the camera axis (not Euclidean distance). Sketch the range image captured by the camera. Calculate the intensities of the vertices seen by the camera.
- 11.2 Derive an equation for the surface of zero disparity shown in Figure 11.3.
- 11.3 Consider the convergent binocular imaging system shown in Figure 11.9. The cameras and all the points are in the $y = 0$ plane. The image planes are perpendicular to their respective camera axes. Find the disparity corresponding to the point P . *Hint:* The perpendicular distance between any point (x_0, z_0) and a line given by $Ax + Bz + C = 0$ is $(Ax_0 + Bz_0 + C)/\sqrt{A^2 + B^2}$.
- 11.4 Consider the binocular stereo imaging system shown in Figure 11.10. Find the disparity, $x_d = |x_1 - x_2|$, for the point P located at $(10, 20, 10)$.

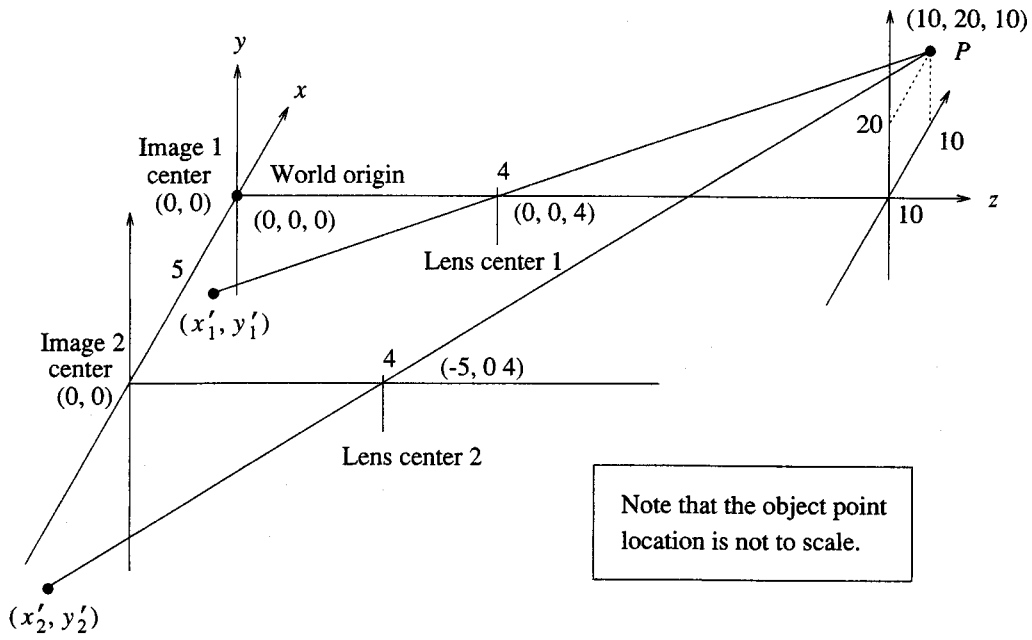


Figure 11.10: Binocular stereo imaging system for Exercise 11.4.