STEREO IMAGE PROCESSING

While pictures acquired using video or still cameras contain a wealth of information about imaged scenes, they reveal little about the world's three-dimensional (3-D) structure and shape. When an object is photographed, its location in the resulting image can be found by tracing a straight line from the object, through the camera's center of projection, and onto the image plane, as is demonstrated in Fig. 1(a). Unfortunately, if you are given only the image location of an object, its actual location in the 3-D world can fall anywhere along that ray, called the *line of sight.* Image-based 3-D measurements are often desired, however, in a multitude of applications ranging from remote mapping to industrial automation, posing the question: how can depth be recovered from photographic images?

One possible answer is provided by our own natural 3-D video systems—our human eyes. Since our eyes are separated by several centimeters, they provide us with two unique viewpoints for our surroundings. Our brains are therefore able to infer the 3-D structure of what we see by determining the relative shift of objects in our retinal images. The following experiment illustrates this principle. Hold your hand a little less than arm's length in front of your face, and alternately close your left and right eyes. Notice two important phenomena: (1) the relative position of your hand and the background depends on which eye is closed, and (2) the perception of depth is limited when only one eye is open.

Stereo, or binocular, image processing systems attempt to recover 3-D structure in the same manner. Just as in retinal imagery, objects photographed from two different locations appear in different image locations. For example, the *stereo pair* of images shown in Fig. 2 was acquired using cameras separated by approximately 1.0 m. Notice that the change in position of the soda can is much larger than that of the textbook. In fact, the can appears to the right of the book in Fig. 2(a) and to its left in Fig. 2(b).

Because this difference in image position, or *disparity,* is uniquely depth-dependent, knowledge of the disparity of ev-

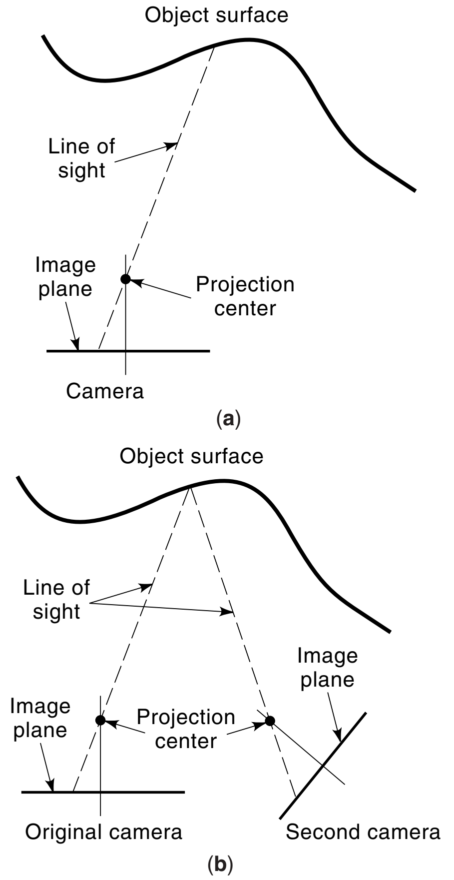

Figure 1. (a) Single-camera imaging geometry. Given only an image location, the object can lie anywhere along the line of sight. (b) Stereo imaging geometry. Given two distinct views, the object's 3-D location is uniquely specified.

ery imaged object, called the *disparity map,* enables stereo vision systems to infer the depth of those objects, assuming the positions of the cameras are known. Remember that for one photographic image, the location of an object in the world can lie anywhere along its line of sight. However, if a second camera is used to photograph the same object, that object must be located along the line of sight emanating from that second camera as well. Since the object must lie somewhere on two nonparallel lines, it must be located at the intersection of those lines [see Fig. 1(b)]. Thus, if the same object can be located in both images and the positions of the cameras are specified, the 3-D position of that object can be computed using trigonometric techniques. This depth recovery process is called *triangulation.*

Hence, two issues must be addressed in order to build a working stereo system. First, algorithms must be developed for automatically locating similar items in both the left and right images, and computing their disparities. This is known as the *correspondence problem.* While people are quite adept at recognizing the same object in two images, the design of computer-based matching algorithms has proven difficult, and is still an active area of engineering research.

After the disparity of image features has been determined, a stereo system must accomplish the task of *reconstruction.* This problem is subdivided into two parts: (1) determining the relative positions of the cameras, and (2) calculating the 3-D world coordinates of image objects using triangulation. Many of the techniques required for reconstruction are, therefore, *calibration procedures.* While a simple camera calibration method is presented in this article as an illustration, general knowledge of this topic facilitates a broader understanding of the stereo reconstruction process.

The remainder of this article provides an in-depth look at three topics: (1) correspondence algorithms, (2) stereo imaging geometry, and (3) 3-D reconstruction, all of which play important roles in the design of stereo image-processing systems. A short discussion of instrumentation for and the future of binocular imaging concludes this work.

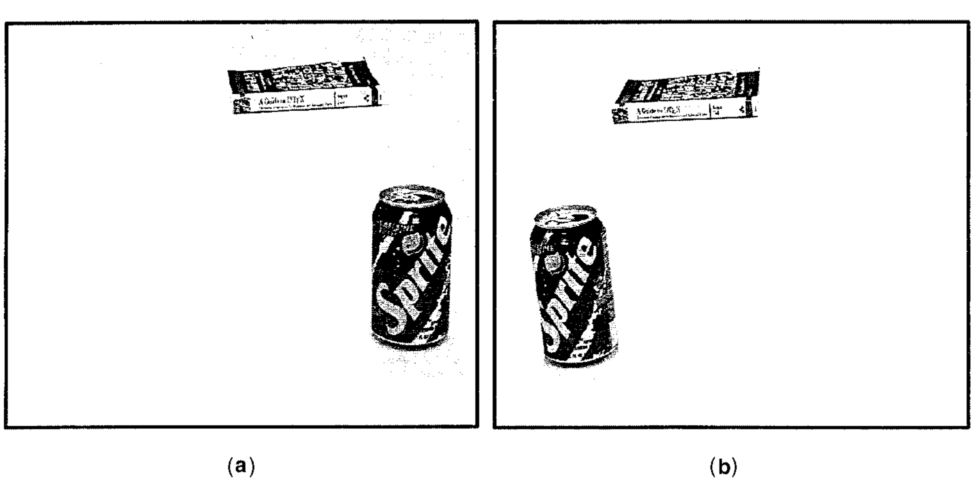

Figure 2. Example of a stereo image pair. Objects at different depths have unique relative shifts between image locations. The closer soda can is located to the right of the book in (a) and to its left in (b).

THE CORRESPONDENCE PROBLEM

Myriad matching techniques have been developed for establishing correspondence in stereo vision systems. These existing techniques are broadly classified into two categories: (1) region-based methods, and (2) feature-based methods. A summary of the most common matching algorithms is presented here; the reader is referred to a survey by Dhond and Aggarwal (1) for further reading regarding approaches for solving the correspondence problem.

In region-based techniques, correspondence is based on the similarity of local image intensity structures. Region-based techniques treat images as two-dimensional signals, employing signal processing and statistical pattern recognition tools to determine correspondence. Since these methods operate on image windows, it is possible to construct very dense disparity, and thus depth, maps of the scene. The accuracy of these methods is, however, affected by random sensor noise and illumination conditions.

Feature-based correspondence methods use local features such as edge points, lines, corners, or other identifiable shapes for matching. These techniques are generally used when scenes contain stable intensity structures that can be reliably and repeatedly extracted from both images, such as edges, corners, and circles. Unlike region-based techniques, these methods use numerical and symbolic properties of features to determine correspondence instead of direct intensity comparisons. For example, criteria for matching lines in two images can be based on any of the following properties: (1) their length, (2) their orientation, or (3) their midpoint coordinates. Pairing features typically provides a more accurate disparity determination, because these feature properties are relatively insensitive to sensor noise and lighting conditions. However, feature-based methods do not typically generate dense disparity maps, since only a small fraction of a typical image contains high-level features.

Region-Based Techniques—Cross-Correlation

Letting (u, v) represent image pixel coordinates, the normalized cross-correlation (2,3) of two $L \times M$ image windows $a(u, v)$ and $b(u, v)$ is given by

$$\phi_{ab} = a \otimes b = \sum_{\chi=0}^{L}\sum_{\nu=0}^{M} \frac{a(u+\chi, v+\nu)b(\chi, \nu)}{a(\chi, \nu)^2 b(\chi, \nu)^2}$$

If $a = b$, then $\phi(a, b) = 1$. As a becomes less and less "similar" to b in structure, $\phi(a, b) \rightarrow 0$. Thus, to determine correspondence using normalized cross-correlation, a window $a(u, v)$ from the left image is compared with various windows $b_i(u, v)$ extracted from a specified region in the right image. The location (u_m, v_m) where the magnitude of the resulting correlation surface is maximal provides the location of the right image window $b_m(u, v)$ that corresponds to the left image window $a(u, v)$. The same technique can be used if the SSD (4), or sum-of-squares difference, given by

$$\psi_{ab} = \sum_{\chi=0}^{L}\sum_{\nu=0}^{M} -[a(u+\chi, v+\nu) - b(u+\chi, v+\nu)]^2$$

is used instead of correlation.

While these processes are effective correspondence tools, both normalized cross-correlation and SSD are very sensitive to photometric variations and sensor noise. The inverses of these functions are also not unique—that is, for a given window $a(u, v)$ there exist windows $b_i(u, v)$, $i = 1, 2, \ldots, M$, such that

$$\phi = a \otimes b_i \quad \forall i = 1, 2, \ldots, M$$
$$\psi = \mathrm{SSD}(a, b_i) \quad \forall i = 1, 2, \ldots, M$$

The maximal magnitude in the correlation or SSD surfaces can therefore appear at a place other than (u_m, v_m), if the windows $b_i(u, v)$ are degraded by noise.

Region-Based Techniques—Cepstral Filtering

Cepstral filtering techniques (5,6) for determining feature correspondence are based on ideas first employed by sonar engineers in the 1960s. Again let $a(u, v)$ be an image window from the left image. Its corresponding image window in the right image can be represented as

$$b_m(u, v) = sh(u-\lambda, v-\nu) * a(u, v)$$

where λ is the image disparity in the horizontal direction, ν is the disparity in the vertical direction, and s is a real constant, $|s| < 1$. The function $h(u, v)$ represents the difference in the point spread functions of the two cameras. From these two windows, the 2-D array $y(u, v)$ can be constructed such that

$$y(u, v) = a(u, v) + b_m(u, v) = a(u, v) + sh(u-\lambda, v-\nu) * a(u, v)$$

The *power cepstrum* of this window is defined as the *inverse z transform of the logarithm of the power spectrum of a sequence.* The log of the power spectrum of $y(u, v)$ is

$$\log|Y(z_u, z_v)|^2 = \log|A(z_u, z_v)|^2 + \sum_{k=1}^{\infty} \frac{-1^{k+1}}{k}[sH(z_u, z_v)z^{-\lambda-\nu}]^k + \sum_{k=1}^{\infty} \frac{-1^{k+1}}{k}[sH^*(z_u, z_v)z^{\lambda+\nu}]^k$$

Thus, the power cepstrum of $y(u, v)$, $\check{y}(u, v) = \mathcal{Z}^{-1}\{\log|Y(z_u, v)|^2\}$ is given by

$$\begin{aligned}\check{y}(u, v) = {} & \check{a}(u, v) + sh(u-\lambda, v-\nu) - \frac{s^2}{2}h(u-2\lambda, v-2\nu) \\ & * h(u-2\lambda, v-2\nu) + \cdots \\ & + sh(-u-\lambda, -v-\nu) - \frac{s^2}{2}h(-u-2\lambda, -v-2\nu) \\ & * h(-u-2\lambda, -v-2\nu) + \cdots\end{aligned}$$

If $y(u, v)$ contains a distortionless echo of $x(u, v)$, then $h(u, v) = \delta(u, v)$, and the convergent series in $\hat{y}(u, v)$ provides a large, discernible peak at $(u = \lambda, v = \nu)$. If the intensity profile $x(u, v)$ appears distorted in the left image, then $h(u, v) \neq \delta[u, v]$, and the energy in the point spread function is dispersed, decreasing the magnitude of the peak at $(u = \lambda, v = \nu)$. In either case, however, the coordinates of the peak in the cepstral array represent the disparity of the chosen image window $a(u, v)$.

To determine the disparity of an image window $a(u, v)$ using cepstral filters, a candidate window $b(u, v)$ must first be chosen that is larger than $a(u, v)$, and is such that it contains the match for $a(u, v)$,

$$b(u,v) = ah(u-\lambda, v-\nu) * a(u,v) + n(u,v)$$

where $n(u, v)$ represents the pixels in b not related to the image of $a(u, v)$. The sequence $y(u, v) = a(u, v) + b(u, v)$ is then formed, and its power cepstrum $\hat{y}(u, v)$ is computed. The vertical and horizontal disparities of $a(u, v)$ are then determined by locating the largest peak in the power cepstrum not including the origin.

The prewhitening provided by cepstral filtering facilitates the establishment of correspondence even if the SNR is low. Cepstral filtering methods, therefore, perform better than other region-based techniques when applied to noisy, low-quality images. However, employing the power cepstrum for large images is computationally inefficient compared to correlation-based methods.

Feature-Based Techniques

The number of methods for determining correspondence using features is enormous. For almost every feature that can be extracted from images, their are a multitude of algorithms for matching them. The vast majority of these algorithms, however, employ the same three-step sequence to determine correspondence:

1. Image feature extraction and parametrization
2. Initial candidate pair selection
3. Relaxation based on similarity/consistency constraints

To further illustrate this process, we describe below a typical feature-based correspondence algorithm known as constrained relaxation (7).

The first step in any feature-based process is to decide what type of features will be used and how they will be extracted from the images. Methods for determining the numerical attributes, or *feature descriptors,* of these features must then be developed. While myriad algorithms have been developed that employ features such as lines, corners, and ellipses, the relaxation method we describe uses the Moravec *interest operator* (8,9) to determine the location of image features due to its applicability in both man-made and natural environments. In general, an interest operator is a nonlinear filter applied to images to detect "interesting" image regions—that is, areas where the intensity variations are classified as exhibiting some sort of desired pattern.

The Moravec operator employs a four-phase, nonlinear process to measure the distinctness of the intensities in a local image region. In the first stage, a variance measure at each pixel $\bar{x} = (u, v)$,

$$\operatorname{var}(\bar{x}) = \left(\sum_{k=0}^{W}\sum_{i=0}^{W}[a(u,v) - a(u+k, v+l)]^2\right)^{1/2}$$

is calculated, where W specifies the size of the local pixel neighborhood. Next, the value of the interest operator $M(\bar{x})$ is the minimum variance of itself and its immediate neighbors:

$$M(\bar{x}) = \min_{\bar{y}\le 1} \operatorname{var}(\bar{x}+\bar{y})$$

where the notation $\bar{x} + \bar{y}$ represents all of the coordinates that are within one pixel of $\bar{x}$. The array $M(\bar{x})$ is then scanned and only local maxima values are retained,

$$M(\bar{x}) = 0 \qquad \text{unless} \quad M(\bar{x}) \ge M(\bar{x}+\bar{y}) \quad \forall \bar{y} \le 1 \tag{1}$$

In the final stage, features of interest are chosen from $M(\bar{x})$ by thresholding. Only those points $\bar{x}$ where $M(\bar{x}) > T$ are kept.

After the candidate features have been extracted and parametrized, initial links between features in one image and potential matches in the other image must be established. In this stage of the constrained relaxation algorithm, each candidate region identified by the Moravec operator in the left image, $a_i(u, v)$ $\forall i = 1, 2, \ldots, M$, is assigned a list of probabilities p_i^j pertaining to the likelihood of its matching each candidate region in the right image, b_j $\forall j = 1, 2, \ldots, N$. The probability list for each candidate $a_i(x, y)$ is initially defined using normalized cross-correlation:

$$p_i^j(0) = a_i(u,v) \otimes b_j(u,v) \qquad \forall j$$

This algorithm initializes itself utilizing the assumption that the stronger the intensity-based match of these features, the greater the likelihood that these features correspond.

Information about the nature of the world is used to refine initial matching probabilities in the final phase of feature-based matching algorithms. The range of possible disparities can be predetermined, for example, if the stereo system will be used only in an indoor environment. The matching algorithm can then reduce the probability of any match producing a disparity greater than this limit to zero, eliminating that pairing. Another possible disparity constraint is related to the smoothness of surfaces. For typical off-the-shelf cameras, it can often be assumed that the spatial resolution of the imaging media is such that disparities of neighboring image points do not generally change drastically. In other words, it can be assumed that depth changes slowly and smoothly on individual objects, and depth discontinuities occur only at the boundaries between objects, such as the table and the floor. Many other restrictions can be imposed on the matching process based on uniqueness, disparity gradient, or contour continuity concepts. These constraints help to resolve ambiguities between different candidate correspondences that have similar initial matching probabilities.

In the constrained relaxation method, the matching probability list for each feature a_i is iteratively refined to impose the smoothness constraint discussed above. In each update cycle, the probability $p_i^j(t)$ is increased if the local neighbors of a_i, $a_l \in a_i \pm \epsilon$, have high probability candidates with disparities similar to that represented by $p_i^j(t)$. Specifically, a quality measure $q_i^j(t)$ is defined for each feature point a_i with neighbors a_l such that

$$q_i^j(t) = \sum_S p_l^j(t)$$

where S is the subset of probabilities $p_l^j(t)$ in a_l that lead to matches with disparities that are close to that represented by $p_i^j(t)$. This quantity enforces the smoothness constraint by increasing in proportion to the number of neighbors of a_i having similar potential disparities of high probability. The probability measure $p_i^j(t)$ is updated at the tth iteration using the relation

$$p_i^j(t+1) = p_i^j(t)[a + bq_i^j(t)]$$

where a and b are user-specified constants that control the speed of convergence. This procedure is repeated until either a prespecified number of iterations has been performed, or the probabilities all reach steady-state values. The pairing (a_i, b_j) that produces the maximum probabilities $p_i^j(t_{\text{final}})$ is returned as the corresponding feature for each a_i.

The Correspondence Problem—Some Final Remarks

There is, unfortunately, no single correspondence method that provides accurate results in all possible imaging environments. Matching features in images is still an open research problem in the field of computer vision. System designers must, therefore, weigh many factors carefully, including but not limited to hardware cost, software development time, lighting conditions, and sensor quality, before choosing a correspondence technique. While there is no "silver bullet" algorithm, there are some generalizations that can be made about the relative abilities of the various techniques.

Region-based methods are generally simple to implement, and special purpose hardware exists enabling these techniques to be executed in real time. They also have the added advantage of generating denser disparity maps than feature-based techniques. However, they require images with significant local intensity contrast, or *texture,* and can be very sensitive to changes in illumination.

Feature-based techniques are much more reliable in systems employing noisy sensors, since feature extraction is less sensitive to illumination changes and random noise. Using a priori knowledge of an application, it is possible to choose optimal feature types, increasing matching performance. For example, if the system is going to be placed in an indoor environment, the acquired stereo images will contain a large number straight lines for matching due to the presence of man-made structures, such as doors, tables, cabinets, and so on. Since these scenes rarely contain significant texture (most people do not paint patterns on their walls), line-based matching algorithms outperform region-based methods indoors. However, only very sparse depth maps can be reconstructed using feature-based disparity maps, and implementation of feature-based algorithms is often very complex.

The performance of these algorithms is enhanced by choosing the proper application-based disparity constraints. Although discussed above in the context of a feature-based method, constraints can be incorporated into the region-based systems as well. While the aforementioned disparity restrictions are useful, knowledge of the imaging system's geometry provides even stricter restraints on the possible locations of image features, as we discuss in the following section.

EPIPOLAR GEOMETRY

A stereo system's imaging geometry, known as its *epipolar geometry,* not only allows for 3-D reconstruction of imaged objects, but also significantly constrains the correspondence process. In fact, an object in one image must lie along a line, called the *epipolar line,* in the other. Therefore, the correspondence search can be reduced to a one-dimensional problem once the epipolar geometry of the system is known.

We assume in this discussion that the reader is familiar with both the pinhole projection camera model and the fundamentals of camera calibration. For further information on these topics, see Ref. 10 or 11.

Epipolar Geometry—Fundamentals

In the general stereo configuration of Fig. 3, the ray representing the line of sight for the left image point $p_l = [u_l\ v_l\ f_l]^T$ is given by

$$\vec{P}_l = [X_l \quad Y_l \quad Z_l]^T = sp_l = s[u_l \quad v_l \quad f_l]^T \tag{2}$$

where s is a scalar. Since the relation between the two camera projection centers is a rigid transformation described by a rotation matrix $\mathbf{R} \in \mathscr{R}^{3\times3}$ and a translation vector $\vec{T} \in \mathscr{R}^3$, the coordinates of this ray in the right camera's reference frame are

$$\vec{P}_r = [X_r Y_r Z_r]^T = s\mathbf{R}p_l + \vec{T}$$

Their projection into the right image yields

$$u_r = \frac{f_r}{Z_r} X_r \quad \text{and} \quad v_r = \frac{f_r}{Z_r} Y_r$$

Assuming that $f_l = f_r = f$, the relation between u_r and v_r is given by

$$u_r = \frac{X_r}{Y_r} v_r$$

Notice that since the $\boldsymbol{P}_r$ is a straight line, the ratio X_r/Y_r is a constant, and thus the relation between u_r and v_r is linear as well. The resulting right image line connects the image of the left camera's center of projection [$s = 0$ in Eq. (2)] to the image of the ray's vanishing point ($s \to \infty$). The point corresponding to $s = 0$, labeled e_r in Fig. 3, is called the *epipole.* A similar linear relation and epipole, e_l, are obtained in the left image for lines of sight emanating from the right camera.

Consider the plane, defined as the *epipolar plane,* formed by the world point and the two projection centers (P, O_l, and O_r in Fig. 3). This plane intersects each image to form the line we recovered above, called the *epipolar line.* Hence, the orien-

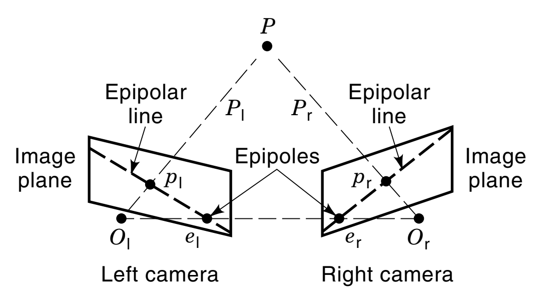

Figure 3. Illustration of general epipolar geometry.

tation of the epipolar plane and corresponding epipolar lines depends only on the location of the world point P, assuming the cameras are stationary. It is interesting to note, however, that all of the epipolar lines will intersect at the epipoles of the images, since each camera's projection center falls along on every possible line of sight by definition.

The practical significance of epipolar geometry is as follows. Given the point p_l, P must lie somewhere along the ray connecting O_l and p_l. Since the points along this ray project onto the epipolar line in the right image, the point p_r that corresponds to point p_l must lie somewhere along that epipolar line. Thus, the search for p_r is confined to the epipolar line, reducing the correspondence problem to one dimension. This constraint can only be employed, however, if the location of the epipolar lines is computed before determining correspondences.

Epipolar Geometry—The Essential Matrix

Let the points P_l and P_r represent the world point P in their respective camera reference frames. Again, since the relative positions of the cameras are related by a rigid transformation,

$$P_r = \mathbf{R}P_l + \vec{T}$$

The equation of the epipolar plane is thus written using the previous equation and the coplanarity condition as

$$(\mathbf{R}^T P_r)^T \vec{T} \times P_l = P_r^T \mathbf{R} \vec{T} \times P_l = 0$$

Recalling from linear algebra the relation for 3×1 vectors

$$\vec{a} \times \vec{b} = \mathbf{A}\vec{b}, \qquad \text{where} \quad \mathbf{A} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_2 \\ -a_2 & a_1 & 0 \end{bmatrix}$$

the equation for the epipolar plane is rewritten, yielding

$$P_r^T \mathbf{RS} P_l = P_r^T \mathbf{E} P_l = 0 \tag{3}$$

where

$$\mathbf{S} = \begin{bmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{bmatrix} \quad \text{and} \quad \mathbf{E} = \mathbf{RS}$$

The matrix $\mathbf{E}$ is called the *essential matrix* because it contains all of the essential information regarding the extrinsic parameters of the stereo geometry.

This essential matrix relation can also be written in terms of image coordinate points $p_i = [u_i, v_i, f]^T$, $i = l, r$. The transformation between the world point P_i, $i = l, r$, and its projected images p_i, $i = l, r$, is

$$p_i = \frac{f}{Z_i} P_i, \qquad i = l, r \tag{4}$$

Through substitution of Eq. (4) and division of both sides by the projective scaling factors f/Z_i, $i = l, r$, Eq. (3) can be rewritten in terms of the image coordinates to yield

$$p_r^T \mathbf{E} p_l = 0 \tag{5}$$

Thus, if the intrinsic camera parameters (focal length, pixel scaling, and image center) are known, then Eq. (5) can be used to recover the essential matrix of the stereo system.

Epipolar Geometry—The Eight-Point Algorithm

There are numerous methods for recovering the essential matrix. While nonlinear methods can be employed that require only five points, the simplest and most often used technique for estimating the relative camera pose is called the *eight-point algorithm,* first reported by Longuet-Higgins (12), and expanded by others (13). Equation (5) can be rewritten in terms of the essential matrix entries as

$$\mathbf{C}\vec{E} = 0 \tag{6}$$

where $\mathbf{C} \in \mathscr{R}^{n\times 9}$ is the system matrix containing corresponding image point coordinates and $\vec{E} \in \mathscr{R}^9$ contains the parameters of the essential matrix. This expression provides one equation in the parameters of $\mathbf{E}$ for every pair of corresponding image points p_i, $i = l, r$. Given $n \geq 8$ corresponding points, a system of these homogeneous equations can be constructed and used to recover a nontrivial solution for the parameters of $\mathbf{E}$. This solution is only unique up to a scale factor, however, due to the system's homogeneity. To recover an exact solution, the actual depth of one point Z_i must be known.

One point of caution regarding this algorithm. The eight-point technique is numerically unstable. Because the image coordinates (u, v) are typically an order of magnitude greater than the focal length, the matrix $\mathbf{C}$ is typically ill conditioned. For further reading regarding methods for coping with this instability, we refer the reader to an article by Hartley (14).

Epipolar Geometry—Recovering the Epipoles and the Epipolar Lines

Once the essential matrix has been recovered, the location of the epipoles in the two images can be determined. Since the pixel location of the epipole in the right image, $\bar{e}_r$, must lie on every possible epipolar line, Eq. (5) can be written as

$$e_r^T \mathbf{E} p_l = 0$$

for every possible point p_l. Since $\mathbf{E} \neq 0$ and $p_l \neq 0$ in general, the above expression implies that

$$e_r^T \mathbf{E} = 0$$

The epipole e_r is therefore the null space of $\mathbf{E}^T$. Using similar logic, the left epipole, e_l must be the null space of $\mathbf{E}$. Therefore, given

$$\text{SVD}(\mathbf{E}) = \mathbf{UDV}^T$$

e_r is given by the column of $\mathbf{U}$ corresponding to the null singular value of $\mathbf{E}$, and e_l is given by the column of $\mathbf{V}$ corresponding to the null singular value of $\mathbf{E}$.

Determining the epipolar lines in one image for a given point in the other is also possible given the essential matrix. Using techniques from projective geometry, it can be shown that the transformation between the image point p_l and its corresponding epipolar line, $\vec{l}_r$, is given by

$$\vec{l}_r = \mathbf{E} p_l$$

Therefore, knowledge of the essential matrix completely specifies a stereo system's epipolar geometry.

Epipolar Geometry—A Final Remark

As was discussed previously, knowledge of a system's epipolar geometry provides a powerful tool for constraining the correspondence problem. Note, however, that the algorithm presented here (and in fact almost all algorithms for recovering epipolar geometry) relies on the identification of a small number of corresponding image points without the aid of geometric constraints. This small collection of points is usually chosen manually to first determine the geometry of the camera system before applying automated correspondence techniques.

3-D SCENE RECONSTRUCTION

The final task of any stereo system is 3-D scene reconstruction. Depending on the amount of *a priori* information available regarding the system's epipolar geometry, different types of 3-D structure are recoverable. Given a disparity map, a unique 3-D reconstruction of the scene can be recovered using the concept of triangulation, if both the extrinsic and intrinsic calibration parameters of the stereo system are known.

It is also possible to recover a type of 3-D scene reconstruction given either partial or no knowledge of the cameras' extrinsic or intrinsic parameters using methods of projective geometry. These algorithms employ a number of concepts that are beyond the scope of this article, and we have omitted them from the following discussion. We refer readers who are interested in this topic to Refs. 10 and 11.

We again assume that the reader is familiar with the basic concepts and standard notation used in the fields of imaging geometry and camera calibration.

Reconstruction—Coplanar Configuration

When the cameras are configured as shown in Fig. 4 with coplanar sensor planes, parallel optical axes, collinear epipolar lines, and projection centers placed at $Z = 0$, developing a relation between the disparity map and the scene's 3-D structure is straightforward. As stated previously, the imaged projection of the world point P in each camera's reference frame

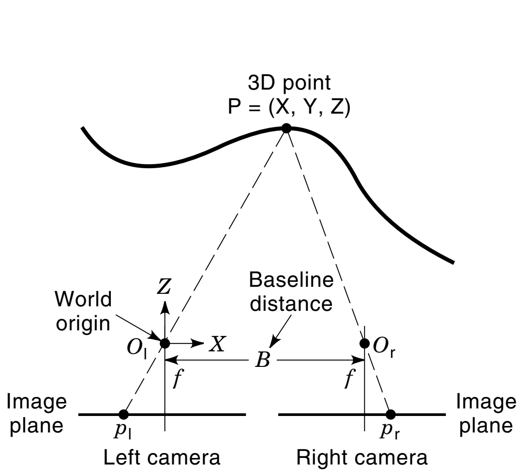

Figure 4. Model of coplanar stereo geometry.

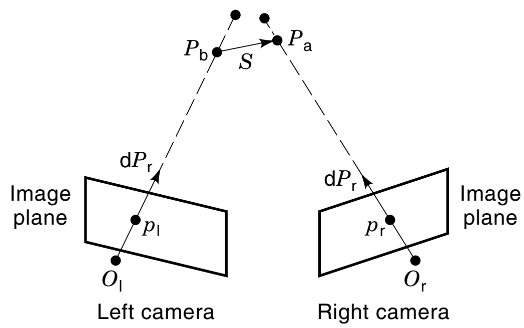

Figure 5. Example of general stereo geometry.

is given by

$$p_i = \frac{f}{Z_i} P_i, \qquad i = \mathrm{l, r}$$

Because the cameras are separated only along the X direction by a distance B, known as the *baseline distance* (see Fig. 5),

$$P_r = P_l + B$$
$$Z_r = Z_l$$

Thus, by substitution

$$P_l = \frac{Z}{f} p_l = \frac{Z}{f} p_r - B$$

Solving this expression for Z yields the following reconstruction relation:

$$Z = \frac{fB}{p_r - p_l} = \frac{fB}{\lambda} \tag{7}$$

where λ is the disparity of the image points p_i, $i = \mathrm{l, r}$. Thus, if the baseline and focal length are known, calculating the depth for corresponding image points is a simple task. For a more detailed discussion of the coplanar stereo geometry, we refer the reader to Ref. 15.

Another important property of coplanar systems is that of horizontal epipolar lines. In this configuration, corresponding feature points will be located in the same row in both images. The correspondence process can therefore be limited to a search along one image row, simplifying software design.

Reconstruction—General Configuration

In the general stereo configuration of Fig. 5, the simple disparity–depth relation given in Eq. (7) does not hold. In addition, determining the intersection of the lines of sight, $\vec{P}_i$, $i =$ l, r, is not trivial. Since the system's epipolar geometry can only be known to within some limited accuracy, these lines of sight might not intersect. The best estimate of P is therefore the midpoint of the vector connecting the two rays at their location of minimum separation.

Let the minimum separation points along the lines of sight be given by $P_a = a\ \overrightarrow{dP}_r$ and $P_b = b\ \overrightarrow{dP}_l$ respectively, where $\overrightarrow{dP}_i$, $i =$ l, r, are the normalized vectors, called *direction cosines,* that point along the lines of sight. Defining $\vec{S}$ as the vector that connects P_a and P_b; then

$$\overrightarrow{dP}_r \cdot \vec{S} = \overrightarrow{dP}_r \cdot (P_a - P_b) = 0$$
$$\overrightarrow{dP}_l \cdot \vec{S} = \overrightarrow{dP}_l \cdot (P_a - P_b) = 0$$

since both rays $\overrightarrow{dP}_i$, $i = \mathrm{l, r}$ are orthogonal to $\vec{S}$. Expanding and using Cramer's rule gives

$$\begin{vmatrix} \overrightarrow{dP}_r \cdot \overrightarrow{dP}_r & -\overrightarrow{dP}_r \cdot \overrightarrow{dP}_l \\ \overrightarrow{dP}_l \cdot \overrightarrow{dP}_r & -\overrightarrow{dP}_l \cdot \overrightarrow{dP}_l \end{vmatrix} \cdot \begin{vmatrix} a \\ b \end{vmatrix} = \begin{vmatrix} \overrightarrow{dP}_r \cdot (O_l - O_r) \\ \overrightarrow{dP}_l \cdot (O_l - O_r) \end{vmatrix} \tag{8}$$

where O_i, $i = \mathrm{l, r}$, are the camera projection centers. Solving Eq. (8) for the scalars a and b yields

$$a = \frac{\begin{vmatrix} \overrightarrow{dP}_r \cdot (O_l - O_r) & -\overrightarrow{dP}_r \cdot \overrightarrow{dP}_l \\ \overrightarrow{dP}_l \cdot (O_l - O_r) & -\overrightarrow{dP}_l \cdot \overrightarrow{dP}_l \end{vmatrix}}{(\overrightarrow{dP}_l \cdot \overrightarrow{dP}_r)(\overrightarrow{dP}_r \cdot \overrightarrow{dP}_l) - \|\overrightarrow{dP}_r\|^2 \|\overrightarrow{dP}_l\|^2}$$

$$b = \frac{\begin{vmatrix} \overrightarrow{dP}_r \cdot \overrightarrow{dP}_r & \overrightarrow{dP}_r \cdot (O_l - O_r) \\ \overrightarrow{dP}_l \cdot \overrightarrow{dP}_r & \overrightarrow{dP}_l \cdot (O_l - O_r) \end{vmatrix}}{(\overrightarrow{dP}_l \cdot \overrightarrow{dP}_r)(\overrightarrow{dP}_r \cdot \overrightarrow{dP}_l) - \|\overrightarrow{dP}_r\|^2 \|\overrightarrow{dP}_l\|^2}$$

The 3-D location of the world point is thus given by the average of the two points,

$$P_w = \frac{P_a + P_b}{2} = \frac{a\,\overrightarrow{dP}_r + b\,\overrightarrow{dP}_l}{2}$$

Unlike coplanar systems, the epipolar lines of a general configuration are not parallel with either image coordinate axis. Thus, even though the search for correspondence is one-dimensional, the desired linear paths are not parallel to the image coordinate axes, making software design more difficult.

Reconstruction—Rectification

It is possible, however, to transform stereo images acquired using a general camera geometry to produce an image pair that appears as if it were taken using a coplanar system. This process, called *rectification,* allows the correspondence problem in general configuration stereo pairs to be restricted to a search along one image row as in coplanar systems.

Rectified images are equivalent to ones that would be obtained if the cameras were rotated around their projection centers until their sensor planes were coplanar as shown in Fig. 6. Rectification algorithms attempt to estimate an image-to-image mapping that simulates the effects of physical camera rotation. In the remainder of this section, we will discuss a three-step rectification technique presented in Refs. 10 and 16.

The first step in this rectification algorithm involves determining a rotation matrix $\mathbf{R}_l$ that makes the left epipole go to infinity. This matrix is constructed using a set of mutually orthogonal unit vectors, $\vec{r}_i \in \mathscr{R}^3$, $i = 1, 2, 3$. If the first vector is chosen as the left epipole, the piercing point assumption above ensures that $\vec{r}_1$ is coincident with the system's transla-

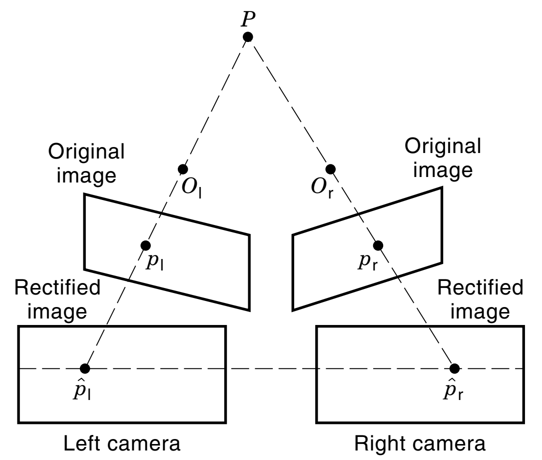

Figure 6. Illustration of the rectification process.

tion direction,

$$\vec{r}_1 = \frac{\vec{T}}{\|\vec{T}\|}$$

Since $\vec{r}_2$ must be orthogonal to $\vec{r}_1$, let $\vec{r}_2$ be defined as the normalized cross product of $\vec{r}_1$ and the optical axis:

$$\vec{r}_2 = (T_x^2 + T_y^2)^{-1/2}[-T_y, T_x, 0]^T$$

The third unit vector is then simply

$$\vec{r}_3 = \vec{r}_1 \times \vec{r}_2$$

The matrix $\mathbf{R}_l$ is thus given by

$$\mathbf{R}_l = \begin{bmatrix} \vec{r}_1^{\,T} \\ \vec{r}_2^{\,T} \\ \vec{r}_3^{\,T} \end{bmatrix} \tag{9}$$

Each point, $p_l = [u_l\ v_l\ f]^T$, in the left image frame is then rotated to form intermediate image points, $\tilde{p}_l$ using the expression

$$\tilde{p}_l = \mathbf{R}_l p_l$$

and then reprojected to form the rectified image points, $\hat{p}_l$, using

$$\hat{p}_l = f/\tilde{z}_l \tilde{p}_l$$

The rectified right image points, $\hat{p}_r$, are then computed using the expressions

$$\begin{aligned} \tilde{p}_r &= \mathbf{R}\mathbf{R}_l p_r \\ \hat{p}_r &= f/\tilde{z}_r \tilde{p}_r \end{aligned} \tag{10}$$

where $\mathbf{R}$ is the platform's actual relative orientation. These rectified images can then be used to determine point correspondence and, if desired, to calculate point depth using the simple coplanar relations. Figure 7 contains an example of the rectification process applied to a typical stereo pair.

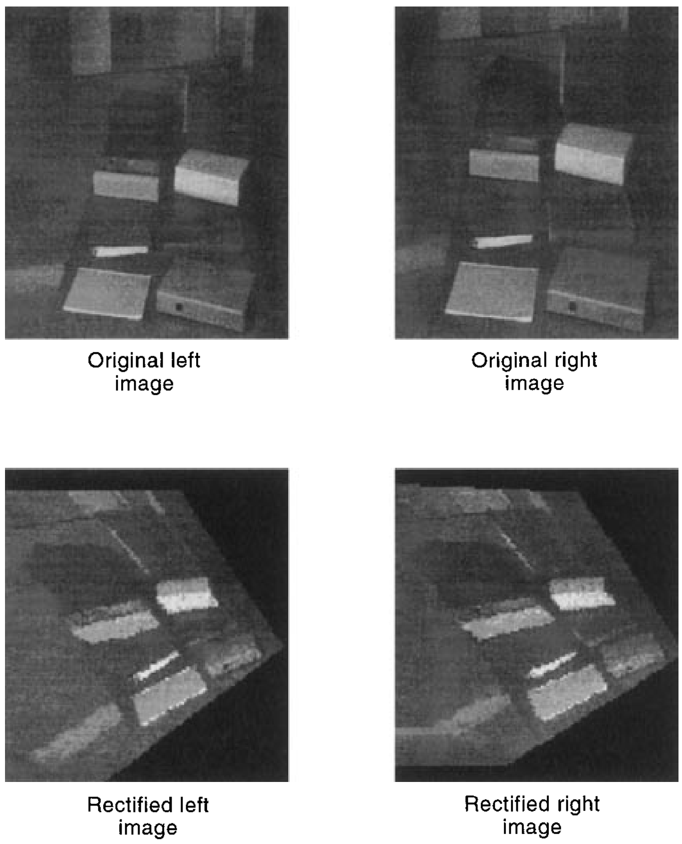

Figure 7. An example of the rectification process applied to a general stereo image pair. Images supplied courtesy of INRIA-Syntim.

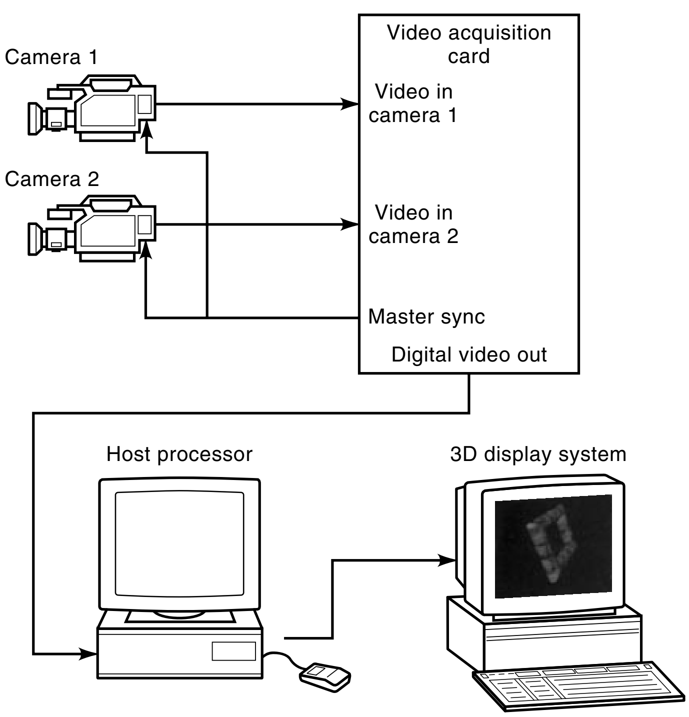

Figure 8. A typical stereo image-processing platform schematic.

INSTRUMENTATION

Since there are no correspondence algorithms or camera configurations that are optimal for every possible task, very few off-the-shelf stereo vision systems are available commercially. Hence, stereo imaging systems are typically designed from components for specific applications. While the choice of a correspondence algorithm is at the core of every design, cost and

Table 1. Manufacturers of Instrumentation for Stereo Imaging

Camera Equipment

- DVC Company
  9450 Mira Mesa Blvd., Suite B 311
  San Diego, CA 92126
  (619) 444-8300
  WWW: www.edt.com/dvc/dvc.html
- Pulnix of America, Inc.
  Mountain View, CA
  (408) 747-0300 ext. 152/127
  WWW: www.pulnix.com
- Eastman Kodak
  Digital Imaging Support Center
  (800) 235-6325
  WWW: www.kodak.com
- Panasonic Industrial Corporation
  Computer Components Group
  6550 Katella Avenue
  Cypress, CA 90630
  (714) 373-7324
  WWW: www.panasonic.com/pic/index-comput.html

Video Acquisition Boards

- Precision Digital Images, Inc.
  8520 154th Avenue NW
  Redmond, WA 98052
  (425) 882-0218
  WWW: www.precisionimages.com
- Coreco, Inc.
  6969 TransCanada, Suite 142
  St. Laurent, PQ H4T 1V8, Canada
  (800) 361-4914
  WWW: www.coreco.com
- Matrox Electronic Systems, Inc.
  1055 St. Regis Blvd.
  Dorval, QC H9P 2T4, Canada
  (800) 804-6243
  WWW: www.matrox.com

Integrated Imaging Systems

- Adept Technology, Inc.
  150 Rose Orchard Way
  San Jose, CA 95134
  (408) 432-0888
  WWW: www.adept.com
- Cognex Corporation
  One Vision Drive
  Natick, MA 01760
  (508) 650-3000
  WWW: www.cognex.com

availability of imaging hardware also play a critical role in stereo design.

A schematic for a standard stereo reconstruction system is shown in Fig. 8. Video cameras are employed to produce pairs of images in either digital or NTSC standard analog format. These images are then transmitted to a video acquisition board, or *framegrabber,* and if necessary redigitized. The host processor next performs the correspondence analysis and scene reconstruction tasks. In the final stage, the 3-D reconstruction of the scene is displayed either by the host or on an external graphics device as shown here. A list of companies that sell imaging-related components is included in Table 1 for readers who want more specific hardware information.

CONCLUDING REMARKS

Stereo image processing is currently a dynamic field that will continue to grow in the near future. Driven by the increased availability of low-cost, high-performance imaging and computational hardware, many engineers are starting to see stereo platforms as a cost-effective method for obtaining real-time 3-D scene or object reconstructions in a growing number of industrial, research, and entertainment applications. For widespread use of stereo vision to become a reality, however, better feature correspondence methods must be developed that are both robust enough to withstand a wide range of noise and illumination conditions, and flexible enough to work with a large number of objects. Despite the myriad new feature-matching and reconstruction techniques reported continually in the research literature, no one has yet been able to demonstrate a high-performance, general-purpose stereo matching scheme. Until this issue is resolved, stereo vision is poised to remain at the forefront of engineering research and scientific inquiry.

BIBLIOGRAPHY

1. U. R. Dhond and J. K. Aggarwal, Structure from stereo—a review, *IEEE Trans. Syst. Man Cybern.,* **20**: 1489–1510, 1989.
2. B. K. Horn, *Robot Vision,* New York: McGraw-Hill, 1991.
3. M. J. Hannah, Bootstrap stereo, *Proc. ARPA Image Understanding Workshop,* College Park, MD, 1980, pp. 201–208.
4. T. Kanade and M. Okutomi, A stereo matching algorithm with an adaptive window, *IEEE Trans. Pattern Anal. Mach. Intell.,* **16**: 920–923, 1994.
5. Y. Yeshurun and E. L. Schwartz, Cepstral filtering on a columnar image architecture: A fast algorithm for binocular stereo segmentation, *IEEE Trans. Pattern Anal. Mach. Intell.,* **11**: 759–767, 1989.
6. T. Olson and D. J. Coombs, Real-time vergence control for binocular robots, *J. Comput. Vision,* **7**: 67–89, 1991.
7. S. T. Barnard and W. B. Thompson, Disparity analysis of images, *IEEE Trans. Pattern Anal. Mach. Intell.,* **2**: 333–340, 1980.
8. D. H. Ballard and C. M. Brown, *Computer Vision,* Englewood Cliffs, NJ: Prentice-Hall, 1982.
9. H. P. Moravec, Towards automatic visual obstacle avoidance, *Proc. 5th IJCAI,* 1977, p. 584.
10. E. Trucco and A. Verri, *Introductory Techniques for 3-D Computer Vision,* Englewood Cliffs, NJ: Prentice-Hall, 1998.
11. O. D. Faugeras, *Three-Dimensional Computer Vision: A Geometric Viewpoint,* Cambridge, MA: MIT Press, 1993.
12. H. C. Longuet-Higgins, A computer algorithm for reconstructing a scene from two projections, *Nature,* **293** (10): 133–135, 1981.
13. R. I. Hartley, Estimation of relative camera positions for uncalibrated cameras, *Proc. 2nd Eur. Conf. Comput. Vision,* Santa Margherita, Italy, 1992, pp. 579–587.
14. R. I. Hartley, In defence of the eight-point algorithm, *Proc. 5th Int. Conf. Comput. Vision,* Cambridge, MA, 1995, pp. 1064–1070.
15. R. C. Gonzalez and R. E. Woods, *Digital Image Processing,* Reading, MA: Addison-Wesley, 1992.
16. N. Ayache, *Artificial Vision for Mobile Robots: Stereo Vision and Multisensory Perception,* Cambridge, MA: MIT Press, 1991.

PHILIP W. SMITH
MONGI A. ABIDI
University of Tennessee

STIMULATION, ELECTRICAL. See HEARING AIDS.

STOCHASTIC ADAPTIVE CONTROL. See STOCHASTIC SYSTEMS.

STOCHASTIC APPROXIMATION. See STOCHASTIC OPTIMIZATION, STOCHASTIC APPROXIMATION AND SIMULATED ANNEALING.

STOCHASTIC OPTIMAL CONTROL. See STOCHASTIC SYSTEMS.