Three-Dimensional Object Recognition from Single Two-Dimensional Images

David G. Lowe

Abstract

A computer vision system has been implemented that can recognize three-dimensional objects from unknown viewpoints in single gray-scale images. Unlike most other approaches, the recognition is accomplished without any attempt to reconstruct depth information bottom-up from the visual input. Instead, three other mechanisms are used that can bridge the gap between the two-dimensional image and knowledge of three-dimensional objects. First, a process of perceptual organization is used to form groupings and structures in the image that are likely to be invariant over a wide range of viewpoints. Second, a probabilistic ranking method is used to reduce the size of the search space during model based matching. Finally, a process of spatial correspondence brings the projections of three-dimensional models into direct correspondence with the image by solving for unknown viewpoint and model parameters. A high level of robustness in the presence of occlusion and missing data can be achieved through full application of a viewpoint consistency constraint. It is argued that similar mechanisms and constraints form the basis for recognition in human vision.

This paper has been published in *Artificial Intelligence*, **31,** 3 (March 1987), pp. 355–395.

1 Introduction

Much recent research in computer vision has been aimed at the reconstruction of depth information from the two-dimensional visual input. An assumption underlying some of this research is that the recognition of three-dimensional objects can most easily be carried out by matching against reconstructed three-dimensional data. However, there is reason to believe that this is not the primary pathway used for recognition in human vision and that most practical applications of computer vision could similarly be performed without bottom-up depth reconstruction. Although depth measurement has an important role in certain visual problems, it is often unavailable or is expensive to obtain. General-purpose vision must be able to function effectively even in the absence of the extensive information required for bottom-up reconstruction of depth or other physical properties. In fact, human vision does function very well at recognizing images, such as simple line drawings, that lack any reliable clues for the reconstruction of depth prior to recognition. This capability also parallels many other areas in which human vision can make use of partial and locally ambiguous information to achieve reliable identifications. This paper presents several methods for bridging the gap between two-dimensional images and knowledge of three-dimensional objects without any preliminary derivation of depth. Of equal importance, these methods address the critical problem of robustness, with the ability to function in spite of missing data, occlusion, and many forms of image degradation.

How is it possible to recognize an object from its two-dimensional projection when we have no prior knowledge of the viewpoint from which we will be seeing it? An important role is played by the process of perceptual organization, which detects groupings and structures in the image that are likely to be invariant over wide ranges of viewpoints. While it is true that the appearance of a three-dimensional object can change completely as it is viewed from different viewpoints, it is also true that many aspects of an object's projection remain invariant over large ranges of viewpoints (examples include instances of connectivity, collinearity, parallelism, texture properties, and certain symmetries). It is the role of perceptual organization to detect those image groupings that are unlikely to have arisen by accident of viewpoint or position. Once detected, these groupings can be matched to corresponding structures in the objects through a knowledge-based matching process. It is possible to use probabilistic reasoning to rank the potential matches in terms of their predicted reliability, thereby focusing the search on the most reliable evidence present in a particular image.

Unfortunately, the matches based on viewpoint-invariant aspects of each object are by their nature only partially reliable. Therefore, they are used simply as "trigger features" to initiate a search process and viewpoint-dependent analysis of each match. A quantitative method is used to simultaneously determine the best viewpoint and object parameter values for fitting the projection of a three-dimensional model to given two-dimensional features. This method allows a few initial hypothesized matches to be extended by making accurate quantitative predictions for the locations of other object features in the image. This provides a highly reliable method for verifying the presence of

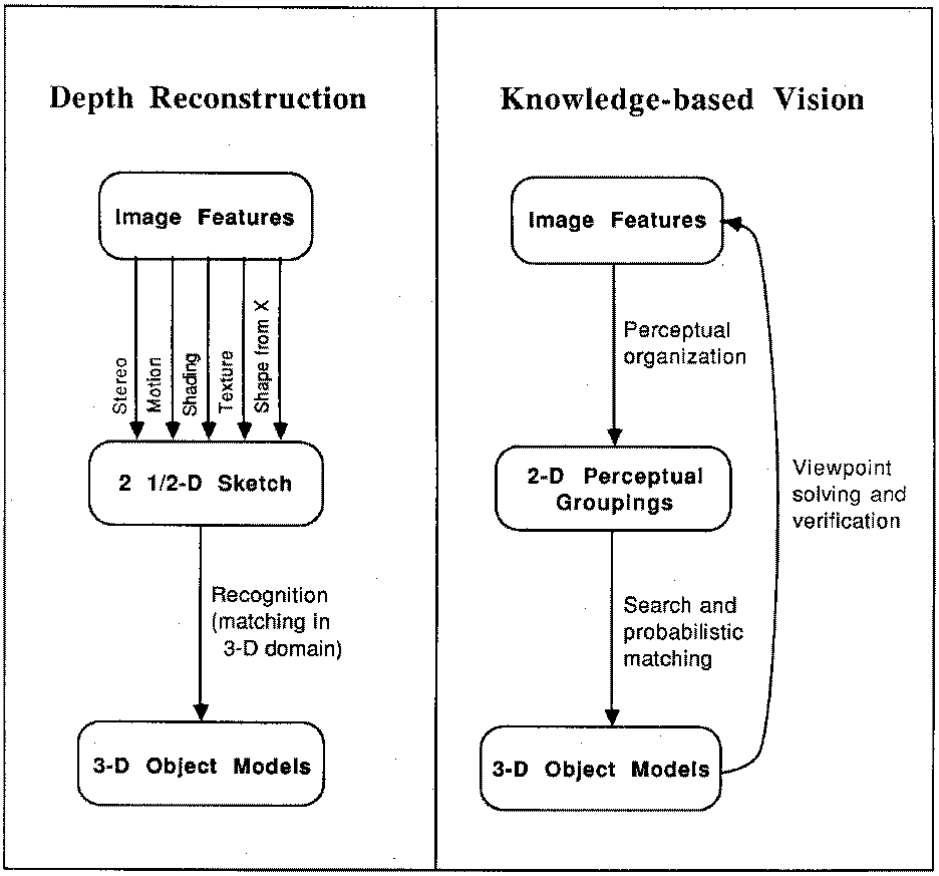

Figure 1: On the left is a diagram of a commonly accepted model for visual recognition based upon depth reconstruction. This paper instead presents the model shown on the right, which makes use of prior knowledge of objects and accurate verification to interpret otherwise ambiguous image data.

a particular object, since it can make use of the spatial information in the image to the full degree of available resolution. The final judgement as to the presence of the object can be based on only a subset of the predicted features, since the problem is usually greatly overconstrained due to the large number of visual predictions from the model compared to the number of free parameters. Figure 1 shows a diagram of these components and contrasts them with methods based upon depth reconstruction.

These methods for achieving recognition have been combined in a fuctioning vision system named SCERPO (for Spatial Correspondence, Evidential Reasoning, and Perceptual Organization). This initial implementation uses simplified components at a number of levels, for example by performing matching only between straight line segments rather than arbitrary curves. However, even this initial system exhibits many forms of robustness, with the ability to identify objects from any viewpoint in the face of partial occlusion, missing features, and a complex background of unrelated image features. The current system has a significant level of performance relative to other

model-based vision systems, including those that are based upon the accurate derivation of depth measurements prior to recognition. In addition, it provides a framework for incorporating numerous potential improvements in image description, perceptual grouping, knowledge indexing, object modeling, and parameter solving, with resulting potential improvements in performance. Many of the components of this system can be related to corresponding capabilities of human vision, as will be described at the relevant points in this paper. The following section examines the psychological validity of the central goal of the system, which is to perform recognition directly from single two-dimensional images.

2 Role of depth reconstruction in human vision

There is a widespread assumption within the computer vision and psychology communities that the recognition of three-dimensional objects is based on an initial derivation of depth measurements from the two-dimensional image [11, 22]. However, this assumption seems to be based more upon the perceived difficulty of achieving recognition from a two-dimensional projection than from any convincing psychological data. In this paper we will argue that human capabilities for recognition are much more general than this restricted model would suggest, and that in fact the need for depth reconstruction is the exception rather than the rule. It is true that human vision contains a number of capabilities for the bottom-up derivation of depth measurements, such as stereo and motion interpretation, and these presumably have important functions. However, biological visual systems have many objectives, so it does not follow that these components are central to the specific problem of visual recognition. In fact, the available evidence would seem to strongly suggest the opposite.

One difficulty with these methods for depth reconstruction is that the required inputs are often unavailable or require an unacceptably long interval of time to obtain. Stereo vision is only useful for objects within a restricted portion of the visual field and range of depths for any given degree of eye vergence, and is never useful for distant objects. At any moment, most parts of a scene will be outside of the limited fusional area. Motion information is available only when there is sufficient relative motion between observer and object, which in practice is also usually limited to nearby objects. Recognition times are usually so short that it seems unlikely that the appropriate eye vergence movements or elapsed time measurements could be taken prior to recognition even for those cases in which they may be useful. Depth measurements from shading or texture are apparently restricted to special cases such as regions of approximately uniform reflectance or regular texture, and they lack the quantitative accuracy or completeness of stereo or motion.

Secondly, human vision exhibits an excellent level of performance in recognizing images—such as simple line drawings—in which there is very little potential for the bottom-up derivation of depth information. Biederman [4] describes an experiment in which almost identical reaction times (about 800 ms) and error rates were obtained for

recognition of line drawings as compared with full-color slides of the same objects from the same viewpoints. Whatever mechanisms are being used for line-drawing recognition have presumably developed from their use in recognizing three-dimensional scenes. The common assumption that line-drawing recognition is a learned or cultural phenomena is not supported by the evidence. In a convincing test of this conjecture, Hochberg and Brooks [15] describe the case of a 19-month-old human baby who had had no previous exposure to any kinds of two-dimensional images, yet was immediately able to recognize ordinary line drawings of known objects. It is true that there has been some research on the bottom-up derivation of depth directly from line drawings or the edges detected in a single image [2, 3, 27], including previous research by the author [20]. However, these methods usually lead to sparse, under-constrained relationships in depth rather than to something resembling Marr's $2\frac{1}{2}$-D sketch. In addition, these methods apply only to special cases and it is often not possible to tell which particular inference applies to a particular case. For example, one often-discussed inference is the use of perspective convergence to derive the orientation of lines that are parallel in three-dimensions; however, given a number of lines in the image that are converging to a common point, there is usually no effective way to distinguish convergence due to perspective effects from the equally common case of lines that are converging to a common point in three-dimensions. In this paper we will make use of many of the same inferences that have previously been proposed for deriving depth, but they will instead be used to generate two-dimensional groupings in the image that are used directly to index into a knowledge base of three-dimensional objects.

Finally, and of most relevance for many applications of computer vision, there has been no clear demonstration of the value of depth information for performing recognition even when it is available. The recognition of objects from complete depth images, such as those produced by a laser scanner, has not been shown to be easier than for systems that begin only with the two-dimensional image. This paper will describe methods for directly comparing the projection of three-dimensional representations to the two-dimensional image without the need for any prior depth information. Since the final verification of an interpretation can be performed by comparing the projected knowledge with each available image to the full accuracy of the data, there is nothing to be gained at this stage from any depth information that is derivative from the original images. The one remaining issue is whether there is some way in which the depth information can significantly speed the search for the correct knowledge to compare to the image.

Of course, none of the above is meant to imply that depth recovery is an unimportant problem or lacks a significant role in human vision. Depth information may be crucial for the initial stages of visual learning or for acquiring certain types of knowledge about unfamiliar structures. It is also clearly useful for making precise measurements as an aid to manipulation or obstacle avoidance. Recognition may sometimes leave the precise position in depth undetermined if the absolute size of an object is unknown. Human stereo vision, with its narrow fusional range for a given degree of eye vergence, seems to be particularly suited to making these precise depth measurements

for selected nearby objects as an aid to manipulation and bodily interaction. However, it seems likely that the role of depth recovery in common instances of recognition has been overstated.

3 Solving for spatial correspondence

Many areas of artificial intelligence are aimed at the interpretation of data by finding consistent correspondences between the data and prior knowledge of the domain. In this paper, we will begin by defining the consistency conditions for judging correspondence between image data and three-dimensional knowledge. Unlike many other areas of artificial intelligence, an important component of this knowledge is quantitative spatial information that requires specific mathematical techniques for achieving correspondence. The particular constraint that we wish to apply can be stated as follows:

> **The viewpoint consistency constraint:** The locations of all projected model features in an image must be consistent with projection from a single viewpoint.

The effective application of this constraint will allow a few initial matches to be bootstrapped into quantitative predictions for the locations of many other features, leading to the reliable verification or rejection of the initial matches. Later sections of this paper will deal with the remaining problems of recognition, which involve the generation of primitive image structures to provide initial matches to the knowledge base and the algorithms and control structures to actually perform the search process.

The physical world is three-dimensional, but a camera's image contains only a two-dimensional projection of this reality. It is straightforward mathematically to describe the process of projection from a three-dimensional scene model to a two-dimensional image, but the inverse problem is considerably more difficult. It is common to remark that this inverse problem is underconstrained, but this is hardly the source of difficulty in the case of visual recognition. In the typical instance of recognition, the combination of image data and prior knowledge of a three-dimensional model results in a highly overconstrained solution for the unknown projection and model parameters. In fact, we must rely upon the overconstrained nature of the problem to make recognition robust in the presence of missing image data and measurement errors. So it is not the lack of constraints but rather their interdependent and non-linear nature that makes the problem of recovering viewpoint and scene parameters difficult. The difficulty of this problem has been such that few vision systems have made use of quantitative spatial correspondence between three-dimensional models and the image. Instead it has been common to rely on qualitative topological forms of correspondence, or else to produce three-dimensional depth measurements that can be matched directly to the model without having to account for the projection process.

Our goal, then, is to carry out a quantitative form of spatial reasoning to provide a two-way link between image measurements and the object model. Matches between

the model and some image features can be used to constrain the three-dimensional position of the model and its components, which in turn leads to further predictions for the locations of model features in the image, leading to more matches and more constraints. The problem of generating the few initial matches to begin this process will be dealt with in later sections of this paper. Here we will describe the spatial reasoning process that relates the image measurements to three-dimensional constraints.

The precise problem we wish to solve is the following: given a set of known correspondences between three-dimensional points on a model and points in a two-dimensional image, what are the values of the unknown projection and model parameters that will result in the projection of the given model points into the corresponding image points. The unknown parameters might include the position and orientation of the object in three dimensions, the focal length of the camera, and various degrees of freedom in the model description, such as articulations or variable dimensions. We will later extend this problem description to allow for the least-squares solution of overdetermined constraints, and to allow for matches between corresponding lines (without concern for the position of endpoints) rather than just points.

There has been some previous work on solving for the position of a rigid three-dimensional object given matches between points on the model and points in the image. This problem must be solved in order to bring photographs into correspondence with mapping data in the field of photogrammetry. An analytic solution, known as the Church method, is presented in the photogrammetry literature [31], but this solution involves nonlinear equations which must themselves be solved by iterative numerical methods. The current preferred technique in photogrammetry is to use the same linearization and iterative methods that will serve as our starting point in this paper. Fischler and Bolles [9] have presented another analytic solution to the problem, but one which also requires iterative numerical solution. In addition, they have presented useful information on the conditions under which multiple solutions can arise. A different approach was taken in the ACRONYM computer vision system [6], which used a general symbolic equation solver to place bounds on projection and model parameters from image measurements. However, the equations for the general projection problem were often much too difficult for exact solution, so sub-optimal bounds would be produced that failed to apply the information inherent in the viewpoint consistency constraint.

The approach taken in this paper will be to linearize the projection equations and apply Newton's method for the necessary number of iterations. A reparameterization of these equations is used to simplify the calculation of the necessary derivatives. This allows us to efficiently solve not only the basic rigid-body problem studied in photogrammetry, but also to extend the method to variable model parameters and forms of correspondence other than the matching of points. Figure 2 illustrates the sequence of steps involved in applying Newton's method to this problem.

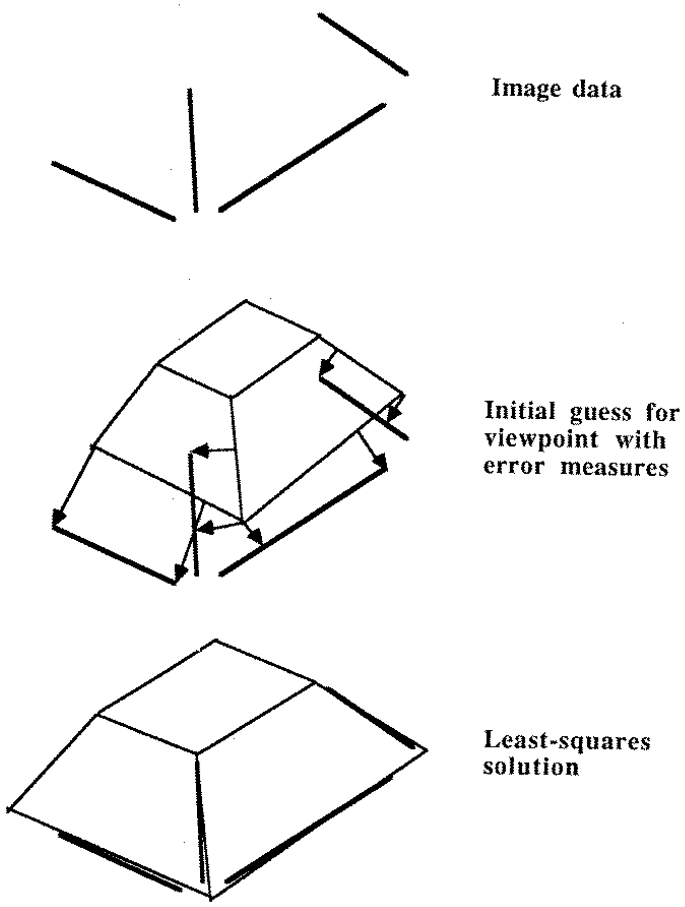

Figure 2: Three steps in the application of Newton's method to achieve spatial correspondence between 2-D image segments and the projection of a 3-D object model.

3.1 Application of Newton's method

Following standard practice in computer graphics, we can describe the projection of a three-dimensional model point $\mathbf{p}$ into a two-dimensional image point (u, v) with the following equations:

$$(x, y, z) = R(\mathbf{p} - \mathbf{t})$$
$$(u, v) = \left(\frac{fx}{z}, \frac{fy}{z}\right)$$

where $\mathbf{t}$ is a 3-D translation vector and R is a rotation matrix which transform $\mathbf{p}$ in the original model coordinates into a point (x, y, z) in camera-centered coordinates. These are combined in the second equation with a parameter f proportional to the camera focal length to perform perspective projection into an image point (u, v).

Our task is to solve for $\mathbf{t}$, R, and possibly f, given a number of model points and their corresponding locations in an image. In order to apply Newton's method, we must

be able to calculate the partial derivatives of u and v with respect to each of the unknown parameters. However, it is difficult to calculate these partial derivatives for this standard form of the projection equation. In particular, this formulation does not describe how to represent the rotation R in terms of its three underlying parameters. Many previous attempts to solve the viewpoint determination problem have treated the rotation as consisting of more than three parameters, which leads to the requirement for more image data than is actually needed and to poor techniques for handling errors.

The partial derivatives with respect to the translation parameters can be most easily calculated by first reparameterizing the projection equations to express the translations in terms of the camera coordinate system rather than model coordinates. This can be described by the following equations:

$$(x, y, z) = R\mathbf{p}$$

$$(u, v) = \left(\frac{fx}{z + D_z} + D_x, \frac{fy}{z + D_z} + D_y \right)$$

Here the variables R and f remain the same as in the previous transform, but the vector t has been replaced by the parameters D_x, D_y and D_z. The two transforms are equivalent [under an affine approximation] when

$$\mathbf{t} = R^{-1} \left[-\frac{D_x(z + D_z)}{f}, -\frac{D_y(z + D_z)}{f}, -D_z \right]^T$$

In the new parameterization, D_x and D_y simply specify the location of the object on the image plane and D_z specifies the distance of the object from the camera. As will be shown below, this formulation makes the calculation of partial derivatives with respect to the translation parameters almost trivial.

We are still left with the problem of representing the rotation R in terms of its three underlying parameters. Our solution to this second problem is based on the realization that the Newton method does not in fact require an explicit representation of the individual parameters. All that is needed is some way to modify the original rotation in mutually orthogonal directions and a way to calculate partial derivatives of image features with respect to these correction parameters. Therefore, we have chosen to take the initial specification of R as given and add to it incremental rotations ϕ_x, ϕ_y and ϕ_z about the x, y and z axes of the current *camera* coordinate system. In other words, we maintain R in the form of a 3x3 matrix rather than in terms of 3 explicit parameters, and the corrections are performed by prefix matrix multiplication with correction rotation matrices rather than by adding corrections to some underlying parameters. It is fast and easy to compose rotations, and these incremental rotations are approximately independent of one another if they are small. The Newton method is now carried out by calculating the optimum correction rotations $\Delta\phi_x$, $\Delta\phi_y$ and $\Delta\phi_z$ to be made about the camera-centered axes. The actual corrections are performed by creating matrices for rotations of the given magnitudes about the respective coordinate axes and composing these new rotations with R.

	x	y	z
ϕ_x	0	$-z$	y
ϕ_y	z	0	$-x$
ϕ_z	$-y$	x	0

Figure 3: The partial derivatives of x, y and z (the coordinates of rotated model points) with respect to counterclockwise rotations ϕ's (in radians) about the coordinate axes.

Another advantage of using the ϕ's as our convergence parameters is that the derivatives of x, y, and z (and therefore of u and v) with respect to them can be expressed in a strikingly simple form. For example, the derivative of x at a point (x, y) with respect to a counter-clockwise rotation of ϕ_z about the z axis is simply $-y$. This follows from the fact that $(x, y, z) = (r\cos\phi_z,\ r\sin\phi_z, z)$, where r is the distance of the point from the z axis, and therefore $\partial x/\partial\phi_z = -r\sin\phi_z = -y$. The table in Figure 3 gives these derivatives for all combinations of variables.

Given this parameterization it is now straightforward to accomplish our original objective of calculating the partial derivatives of u and v with respect to each of the original camera parameters. For example, our new projection transform above tells us that:

$$u = \frac{fx}{z + D_z} + D_x$$

so

$$\frac{\partial u}{\partial D_x} = 1$$

Also,

$$\frac{\partial u}{\partial \phi_y} = \frac{f}{z + D_z}\frac{\partial x}{\partial \phi_y} - \frac{fx}{(z + D_z)^2}\frac{\partial z}{\partial \phi_y}$$

but, from the table in Figure 3, we know that

$$\frac{\partial x}{\partial \phi_y} = z \text{ and } \frac{\partial z}{\partial \phi_y} = -x$$

and, for simplicity, we will substitute

$$c = \frac{1}{z + D_z}$$

giving,

$$\frac{\partial u}{\partial \phi_y} = fcz + fc^2x^2 = fc(z + cx^2)$$

	u	v
D_x	1	0
D_y	0	1
D_z	$-fc^2x$	$-fc^2y$
ϕ_x	$-fc^2xy$	$-fc(z+cy^2)$
ϕ_y	$fc(z+cx^2)$	fc^2xy
ϕ_z	$-fcy$	fcx
f	cx	cy

Figure 4: The partial derivatives of u and v with respect to each of the camera viewpoint parameters.

Similarly,

$$\frac{\partial u}{\partial \phi_z} = \frac{f}{z + D_z}\frac{\partial x}{\partial \phi_z} = -fcy.$$

All the other derivatives can be calculated in a similar way. The table in Figure 4 gives the derivatives of u and v with respect to each of the seven parameters of our camera model, again substituting $c = (z + D_z)^{-1}$ for simplicity.

Our task on each iteration of the multi-dimensional Newton convergence will be to solve for a vector of corrections

$$\mathrm{h} = [\Delta D_x, \Delta D_y, \Delta D_z, \Delta \phi_x, \Delta \phi_y, \Delta \phi_z]$$

If the focal length is unknown, then Δf would also be added to this vector. Given the partial derivatives of u and v with respect to each variable parameter, the application of Newton's method is straightforward. For each point in the model which should match against some corresponding point in the image, we first project the model point into the image using the current parameter estimates and then measure the error in its position compared to the given image point. The u and v components of the error can be used independently to create separate linearized constraints. For example, making use of the u component of the error, E_u, we create an equation which expresses this error as the sum of the products of its partial derivatives times the unknown error-correction values:

$$\frac{\partial u}{\partial D_x}\Delta D_x + \frac{\partial u}{\partial D_y}\Delta D_y + \frac{\partial u}{\partial D_z}\Delta D_z + \frac{\partial u}{\partial \phi_x}\Delta \phi_x + \frac{\partial u}{\partial \phi_y}\Delta \phi_y + \frac{\partial u}{\partial \phi_z}\Delta \phi_z = E_u$$

Using the same point we create a similar equation for its v component, so for each point correspondence we derive two equations. From three point correspondences we can de-

rive six equations and produce a complete linear system which can be solved for all six camera model corrections. After each iteration the corrections should shrink by about one order of magnitude, and no more than a few iterations should be needed even for high accuracy.

Unknown model parameters, such as variable lengths or angles, can also be incorporated. In the worst case, we can always calculate the partial derivatives with respect to these parameters by using standard numerical techniques that slightly perturb the parameters and measure the resulting change in projected locations of model points. However, in most cases it is possible to specify the three-dimensional directional derivative of model points with respect to the parameters, and these can be translated into image derivatives through projection. Examples of the solution for variable model parameters simultaneously with solving for viewpoint have been presented in previous work [18].

In most applications of this method we will be given more correspondences between model and image than are strictly necessary, and we will want to perform some kind of best fit. In this case the Gauss least-squares method can easily be applied. The matrix equations described above can be expressed more compactly as

$$J\mathbf{h} = \mathbf{e}$$

where J is the Jacobian matrix containing the partial derivatives, $\mathbf{h}$ is the vector of unknown corrections for which we are solving, and $\mathbf{e}$ is the vector of errors measured in the image. When this system is overdetermined, we can perform a least-squares fit of the errors simply by solving the corresponding normal equations:

$$J^T J\mathbf{h} = J^T\mathbf{e}$$

where $J^T J$ is square and has the correct dimensions for the vector $\mathbf{h}$.

3.2 Making use of line-to-line correspondences

Another important extension to the basic algorithm is to allow it to use line-to-line correspondences in addition to point-to-point ones. This is important in practice because low-level vision routines are relatively good at finding the transverse locations of lines but are much less certain about exactly where the lines terminate. Line terminations may also be widely displaced due to occlusion, shadows, or various sources of failure in the low-level edge detection algorithms. Therefore, we should express our errors in terms of the distance of one line from another, rather than in terms of the error in the locations of points. The solution is to measure as our errors the perpendicular distance of selected points on the model line from the corresponding line in the image, and to then take the derivatives in terms of this distance rather than in terms of u and v.

In order to express the perpendicular distance of a point from a line it is useful to first express the image line as an equation of the following form, in which m is the slope:

$$\frac{-m}{\sqrt{m^2+1}}\,u + \frac{1}{\sqrt{m^2+1}}\,v = d$$

In this equation d is the perpendicular distance of the line from the origin. If we substitute some point (u', v') into the left side of the equation and calculate the new value of d for this point (call it d'), then the perpendicular distance of this point from the line is simply $d - d'$. What is more, it is easy to calculate the derivatives of d' for use in the convergence, since the derivatives of d' are just a linear combination of the derivatives of u and v as given in the above equation, and we already know how to calculate the u and v derivatives from the solution given for using point correspondences. The result is that each line-to-line correspondence we are given between model and image gives us two equations for our linear system—the same amount of information that is conveyed by a point-to-point correspondence. Figure 2 illustrates the measurement of these perpendicular errors between matching model lines and image lines.

The same basic method could be used to extend the matching to arbitrary curves rather than just straight line segments. In this case, we would assume that the curves are locally linear and would minimize the perpendicular separation at selected points as in the straight line case, using iteration to compensate for non-linearities. For curves that are curving tightly with respect to the current error measurements, it would be necessary to match points on the basis of orientation and curvature in addition to simple perpendicular projection. However, the current implementation of SCERPO is limited to the matching of straight line segments, so these extensions to include the matching of arbitrary curves remain to be implemented.

3.3 The use of parameter determination for matching

The mathematical methods for parameter determination presented above need to be integrated into a matching algorithm that can extend a few initial matches and return a reliable answer as to the presence or absence of the object at the hypothesized location. Some of the implementation details that must be worked out include the choice of a representation for object models, the calculation of a starting viewpoint to initiate Newton iteration, and the methods for making use of the initial parameter determination to generate new matches between image and object features.

The object models used in the current implementation of SCERPO consist simply of a set of 3-D line segments. A primitive form of hidden line elimination is performed by attaching a visibility specification to each line segment. The visibility specification contains a boolean combination of hemispheres from which the segment is visible. Each hemisphere of directions is represented by a unit vector, so that the visibility of the segment can be very efficiently computed by simply checking the sign of the dot product of this vector with the vector pointing to the camera position. It is important that the visibility calculation be fast, as it is performed in the inner loop of the matching process. However, this form of hidden line elimination is only an approximation, since

it does not allow for partial occlusion of a line segment or for calculating occlusion between objects. It would also need to be extended to allow for models with variable internal parameters. As with other parts of the matching process, we rely upon the overall robustness of the system in the face of missing or extra features to compensate for occasional errors in hidden-line determination.

As with all applications of Newton's method, convergence of the viewpoint solving algorithm is guaranteed only for starting positions that are "sufficiently close" to the final solution. Fortunately, in this problem several of the parameters (scaling and translation in the image plane) are exactly linear, while the other parameters (rotation and perspective effects) are approximately linear over wide ranges of values. In practice, we have found that as long as the orientation parameters are within 60 degrees of their correct values, almost any values can be chosen for the other parameters. Reasonable estimates of the viewpoint parameters can be easily computed from the same matches that will be used for convergence. Since most parts of the model are only visible from a limited range of viewpoints, we can select an orientation in depth that is consistent with this range for each object feature being matched. Orientation in the image plane can be estimated by causing any line on the model to project to a line with the orientation of the matching line in the image. Similarly, translation can be estimated by bringing one model point into correspondence with its matching image point. Scale (i.e., distance from the camera) can be estimated by examining the ratios of lengths of model lines to their corresponding image lines. The lines with the minimum value of this ratio are those that are most parallel to the image plane and have been most completely detected, so this ratio can be used to roughly solve for scale under this assumption. These estimates for initial values of the viewpoint parameters are all fast to compute and produce values that are typically much more accurate than needed to assure convergence. Further improvements in the range of convergence could be achieved through the application of standard numerical techniques, such as damping [8].

A more difficult problem is the possible existence of multiple solutions. Fischler and Bolles [9] have shown that there may be as many as four solutions to the problem of matching three image points to three model points. Many of these ambiguities will not occur in practice, given the visibility constraints attached to the model, since they involve ranges of viewpoints from which the image features would not be visible. Also, in most cases the method will be working with far more than the minimum amount of data, so the overconstrained problem is unlikely to suffer from false local minima during the least-squares process. However, when working with very small numbers of matches, it may be necessary to run the process from several starting positions in an attempt to find all of the possible solutions. Given several starting positions, only a couple of iterations of Newton's method on each solution are necessary to determine whether they are converging to the same solution and can therefore be combined for subsequent processing. Finally, it should be remembered that the overall system is designed to be robust against instances of missing data or occlusion, so an occasional failure of the viewpoint determination should lead only to an incorrect rejection of a single match and an increase in the amount of search rather than a final failure in recognition.

3.4 Extending the initial set of matches

Once the initial set of hypothesized matches has been used to solve for the viewpoint parameters, this estimate of viewpoint can be used to predict the locations of other model features in the image and thereby extend the match. The predictions of these image locations can be obtained by simply projecting the model edges onto the image using the current set of viewpoint parameters. The more difficult problem is to select matches in the image that are consistent with these predictions. Rather than setting arbitrary thresholds on the error range within which image segments must fall to be considered in agreement with a predicted model segment, a probabilistic approach is taken. First, each segment is projected, and the image data structure is searched to select potential matches. All line segments in the image are indexed according to location and orientation, so this search can be carried out efficiently. Each potential match is assigned a value giving the probability that a randomly placed image segment would agree with the prediction to within the measured difference in length, orientation and trasverse position. The probability calculation uses the same assumptions and formalisms as are used for the perceptual organization evaluations to be described in the next section. The lower this probability of accidental agreement, the more likely it is that the match is correct.

After evaluating each potential match for a given prediction, the top-ranked match is assigned a higher probability of being mistaken if the second-ranked match has a similar evaluation. The purpose of this penalty is to avoid committing ourselves to either choice of an ambiguous match if there is some less ambiguous alternative from some other model prediction. At each stage of the iteration we select only matches with a probability of being accidental of less than 0.01, or else we select the single lowest probability match from all of the model predictions. These are then added to the least-squares solution to update the estimate of viewpoint. By the time a number of the most reliable matches have been found, the viewpoint estimation will be based on a substantial amount of data and should be accurate enough to choose correctly between the more ambiguous alternatives. The set of matches is repeatedly extended in this way until no more can be found. This iterative matching procedure has the appealing property of using the easy cases to provide better viewpoint estimates to disambiguate the more difficult situations, yet it avoids the expense of search and backtracking.

The final accuracy of the viewpoint determination procedure could probably be improved by attempting to discard the points that deviate the most from the least-squares solution and reconverging on the remaining data. This would allow the system to converge on a consensus viewpoint estimate that was not influenced by the largest errors in modeling or feature detection. However, this procedure remains to be implemented.

The final judgement as to the presence of the object is based simply on the degree to which the final viewpoint estimate is overconstrained. Since only three edges are needed to solve for viewpoint, each further match adds to the verification of the presence of the object. In addition, the least-squares solution provides an estimate of the standard deviation of the error. Given sufficiently detailed models, correct instances of

recognition should be greatly overconstrained even in the face of partial occlusion and other missing features, so the precise threshold for rejection should be unimportant. In the examples that are presented at the end of this paper, correct matches typically had over 20 image segments in close agreement with the model, while incorrect matches seldom found more than 7 matching segments.

4 Perceptual organization

The methods for achieving spatial correspondence presented in the previous section enforce the powerful constraint that all parts of an object's projection must be consistent with a single viewpoint. This constraint allows us to bootstrap just a few initial matches into a complete set of quantitative relationships between model features and their image counterparts, and therefore results in a reliable decision as to the correctness of the original match. The problem of recognition has therefore been reduced to that of providing tentative matches between a few image features and an object model. The relative efficiency of the viewpoint solution means that only a small percentage of the proposed matches need to be correct for acceptable system performance. In fact, when matching to a single, rigid object, one could imagine simply taking all triplets of nearby line segments in the image and matching them to a few sets of nearby segments on the model. However, this would clearly result in unacceptable amounts of search as the number of possible objects increases, and when we consider the capabilities of human vision for making use of a vast database of visual knowledge it is obvious that simple search is not the answer. This initial stage of matching must be based upon the detection of structures in the image that can be formed bottom-up in the absence of domain knowledge, yet must be of sufficient specificity to serve as indexing terms into a database of objects.

Given that we often have no prior knowledge of viewpoint for the objects in our database, the indexing features that are detected in the image must reflect properties of the objects that are at least partially invariant with respect to viewpoint. This means that it is useless to look for features at particular sizes or angles or other properties that are highly dependent upon viewpoint. A second constraint on these indexing features is that there must be some way to distinguish the relevant features from the dense background of other image features which could potentially give rise to false instances of the structure. Through an accident of viewpoint or position, three-dimensional elements that are unrelated in the scene may give rise to seemingly significant structures in the image. Therefore, an important function of this early stage of visual grouping is to distinguish as accurately as possible between these accidental and significant structures. We can summarize the conditions that must be satisfied by perceptual grouping operations as follows:

> **The viewpoint invariance condition:** Perceptual features must remain stable over a wide range of viewpoints of some corresponding three-dimensional structure.

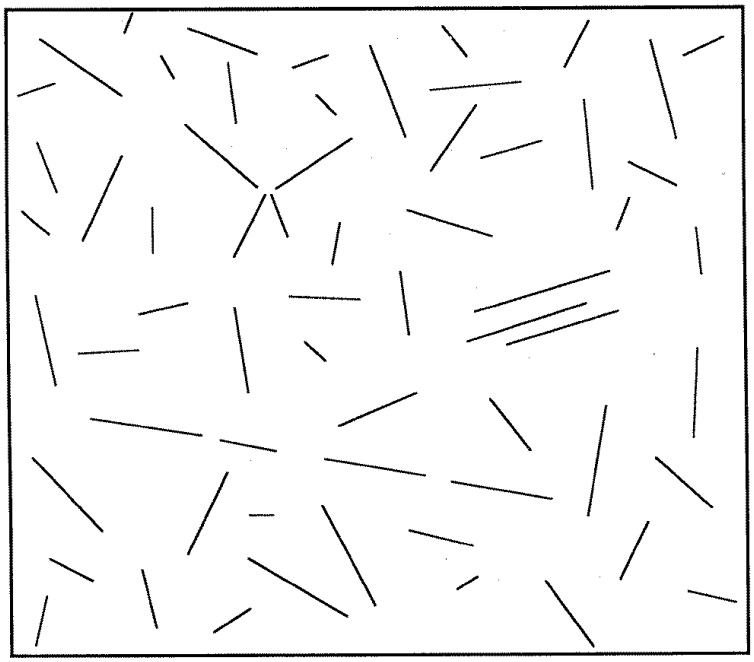

Figure 5: This figure illustrates the human ability to spontaneously detect certain groupings from among an otherwise random background of similar elements. This figure contains three non-random groupings resulting from parallelism, collinearity, and endpoint proximity (connectivity).

> **The detection condition:** Perceptual features must be sufficiently constrained so that accidental instances are unlikely to arise.

Although little-studied by the computational vision community, the perceptual organization capabilities of human vision seem to exhibit exactly these properties of detecting viewpoint invariant structures and calculating varying degrees of significance for individual instances. These groupings are formed spontaneously and can be detected immediately from among large numbers of individual elements. For example, people will immediately detect certain instances of clustering, connectivity, collinearity, parallelism, and repetitive textures when shown a large set of otherwise randomly distributed image elements (see Figure 5). This grouping capability of human vision was studied by the early Gestalt psychologists [29] and is also related to research in texture description [21, 32]. Unfortunately, this important component of human vision has been missing from almost all computer vision systems, presumably because there has been no clear computational theory for the role of perceptual organization in the overall fuctioning of vision.

A basic goal underlying research on perceptual organization has been to discover some principle that could unify the various grouping phenomena of human vision. The Gestaltists thought that this underlying principle was some basic ability of the human mind to proceed from the whole to the part, but this lacked a computational or predictive formulation. Later research summarized many of the Gestaltists' results with the

observation that people seem to perceive the simplest possible interpretation for any given data [14]. However, any definition of simplicity has depended entirely on the language that is used for description, and no single language has been found to encompass the range of grouping phenomena. Greater success has been achieved by basing the analysis of perceptual organization on a functional theory which assumes that the purpose of perceptual organization is to detect stable image groupings that reflect actual structure of the scene rather than accidental properties [30]. This parallels other areas of early vision in which a major goal is to identify image features that are stable under changes in imaging conditions.

4.1 Derivation of grouping operations

Given these functional goals, the computational specification for perceptual organization is to differentiate groupings that arise from the structure of a scene from those that arise due to accidents of viewpoint or positioning. This does not lead to a single metric for evaluating the significance of every image grouping, since there are many factors that contribute to estimating the probability that a particular grouping could have arisen by accident. However, by combining these various factors and making use of estimates of prior probabilities for various classes of groupings, it is possible to derive a computational account for the various classes of grouping phenomena. An extensive discussion of these issues has been presented by the author in previous work [19], but here we will examine the more practical question of applying these methods to the development of a particular vision system. We will simplify the problem by looking only at groupings of straight line segments detected in an image and by considering only those groupings that are based upon the properties of proximity, parallelism, and collinearity.

A strong constraint on perceptual organization is provided by the viewpoint invariance condition, since there are only a relatively few types of two-dimensional image relations that are even partially invariant with respect to changes in viewpoint of a three-dimensional scene. For example, it would be pointless to look for lines that form a right angle in the image, since even if it is common to find lines at right angles in the three-dimensional scene they will project to right angles in the image only from highly restricted viewpoints. Therefore, even if an approximate right angle were detected in the image, there would be little basis to expect that it came from a right angle in the scene as opposed to lines at any other three-dimensional angle. Compare this to finding lines at a 180 degree angle to one another (i.e., that are collinear). Since collinear lines in the scene will project to collinear lines in the image from virtually all viewpoints, we can expect many instances of collinearity in the image to be due to collinearity in three dimensions. Likewise, proximity and parallelism are both preserved over wide ranges of viewpoint. It is true that parallel lines in the scene may converge in the image due to perspective, but many instances of parallelism occupy small visual angles so that the incidence of approximate parallelism in the image can be expected to be much higher than simply those instances that arise accidentally. In summary, the requirement of partial invariance with respect to changes in viewpoint greatly restricts the classes of image

relations that can be used as a basis for perceptual organization.

If we were to detect a perfectly precise instance of, say, collinearity in the image, we could immediately infer that it arose from an instance of collinearity in the scene. That is because the chance of perfect collinearity arising due to an accident of viewpoint would be vanishingly small. However, real image measurements include many sources of uncertainty, so our estimate of significance must be based on the degree to which the ideal relation is achieved. The quantitative goal of perceptual organization is to calculate the probability that an image relation is due to actual structure in the scene. We can estimate this by calculating the probability of the relation arising to within the given degree of accuracy due to an accident of viewpoint or random positioning, and assuming that otherwise the relation is due to structure in the scene. This could be made more accurate by taking into account the prior probability of the relation occurring in the scene through the use of Bayesian statistics, but this prior probability is seldom known with any precision.

4.2 Grouping on the basis of proximity

We will begin the analysis of perceptual organization by looking at the fundamental image relation of proximity. If two points are close together in the scene, then they will project to points that are close together in the image from all viewpoints. However, it is also possible that points that are widely separated in the scene will project to points arbitrarily close together in the image due to an accident of viewpoint. Therefore, as with all of the cases in which we attempt to judge the significance of perceptual groupings, we will consider a grouping to be significant only to the extent that it is unlikely to have arisen by accident.

An important example of the need to evaluate proximity is when attempting to form connectivity relations between line segments detected in an image. The proximity of the endpoints of two line segments may be due to the fact that they are connected or close together in the three-dimensional scene, or it may be due to a simple accident of viewpoint. We must calculate for each instance of proximity between two endpoints the probability that it could have arisen from unrelated lines through an accident of viewpoint. Since we often have no prior knowledge regarding the scene and since the viewpoint is typically unrelated to the structure of the three-dimensional objects, there is little basis for picking a biased background distribution of image features against which to judge significance. Therefore, this calculation will be based upon the assumption of a background of line segments that is uniformly distributed in the image with respect to orientation, position, and scale.

Given these assumptions, the expected number of endpoints, N, within a radius r of a given endpoint is equal to the average density of endpoints per unit area, d, multiplied by the area of a circle with radius r (see Figure 6):

$$N = d\pi r^2$$

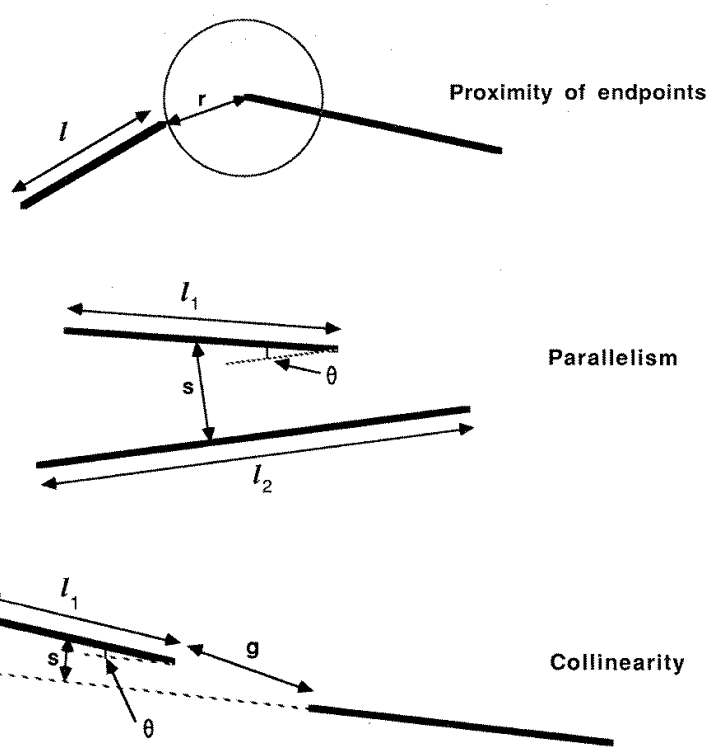

Figure 6: Measurements that are used to calculate the probablity that instances of proximity, parallelism, or collinearity could arise by accident from randomly distributed line segments.

For values of N much less than 1, the expected number is approximately equal to the probability of the relation arising accidentally. Therefore, significance varies inversely with N. It also follows that significance is inversely proportional to the square of the separation between the two endpoints.

However, the density of endpoints, d, is not independent of the length of the line segments that are being considered. Assuming that the image is uniform with respect to scale, changing the size of the image by some arbitrary scale factor should have no influence on our evaluation of the density of line segments of a given length. This scale independence requires that the density of lines of a given length vary inversely according to the square of their length, since halving the size of an image will decrease its area by a factor of 4 and decrease the lengths of each segment by a factor of 2. The same result can be achieved by simply measuring the proximity r between two endpoints as proportional to the length of the line segments which participate in the relation. If the two line segments are of different lengths, the higher expected density of the shorter segment will dominate that of the longer segment, so we will base the calculation on the minimum of the two lengths. The combination of these results leads to the following evaluation metric. Given a separation r between two endpoints belonging to line segments of minimum length l:

$$N = \frac{2D\pi r^2}{l^2}$$

We are still left with a unitless constant, D, specifying the scale-independent density of line segments (the factor 2 accounts for the fact that there are 2 endpoints for each line segment). Since the measures of significance will be used mostly to rank groupings during the search process, the value chosen for a constant factor is of little importance because it will have no influence on the rankings. However, for our experiments we have somewhat arbitrarily assigned D the value 1. In fact, given a fairly dense set of segments with independent orientations and positions, and given the constraint that they do not cross one another, this scale independent measure will have a value close to 1.

This formula does a reasonable job of selecting instances of endpoint proximity that seem perceptually significant. Our concern with uniformity across changes in scale has had an important practical impact on the algorithm. It means that the algorithm will correctly pick out large-scale instances of connectivity between long segments, even if there are many short segments nearby which would otherwise mask the instance of endpoint proximity. This capability for detecting groupings at multiple scales is an important aspect of perceptual organization in human vision.

4.3 Grouping on the basis of parallelism

A similar measure can be used to decide whether an approximate instance of parallelism between two lines in the image is likely to be non-accidental in origin. Let l_1 be the length of the shorter line and l_2 be the length of the longer line. In order to measure the average separation s between the two lines, we calculate the perpendicular distance from the longer line to the midpoint of the shorter line. As in the case for evaluating proximity, we assume that the density of line segments of length greater than l_1 is $d = D/l_1^2$, for a scale-independent constant D. Then, the expected number of lines within the given separation of the longer line will be the area of a rectangle of length l_2 and width $2s$ multiplied by the density of lines of length at least l_1. Let θ be the magnitude of the angular difference in radians between the orientations of the two lines. Assuming a uniform distribution of orientations, only $2\theta/\pi$ of a set of lines will be within orientation θ of a given line. Therefore, the expected number of lines within the given separation and angular difference will be:

$$E = \left(\frac{2sl_2D}{l_1^2}\right)\left(\frac{2\theta}{\pi}\right) = \frac{4D\theta sl_2}{\pi l_1^2}$$

As in the previous case, we assign D the value 1 and assume that significance is inversely proportional to E.

4.4 Grouping on the basis of collinearity

Measuring the probability that an instance of collinearity has arisen by accident shares many features in common with the case of parallelism. In both cases, the ideal relation would involve two line segments with the same orientation and with zero separation perpendicular to the shared orientation. However, in the case of parallelism the line segments are presumed to overlap in the direction parallel to their orientation, whereas in collinearity the segments are expected to be separated along the direction of their orientation with an intervening gap. Let g be the size of this gap (the separation of the endpoints). As in the case of parallelism, let s be the perpendicular distance from the midpoint of the shorter line segment, l_1, to the extension of the longer line segment, l_2. These bounds determine a rectangular region of length $g + l_1$ and width $2s$ within which other lines would have the same degree of proximity. Therefore, by analogy in other respects with the case of parallelism, we get

$$E = \frac{4D\theta s(g + l_1)}{\pi l_1^2}$$

Notice that this measure is independent of the length of the longer line segment, which seems intuitively correct when dealing with collinearity.

4.5 Implementation of the grouping operations

The subsections above have presented methods for calculating the significance of selected relationships between given pairs of straight line segments. The most obvious way to use these to detect all significant groupings in the image would be to test every pair of line segments and retain only those pairs which have high levels of significance. However, the complexity of this process would be $O(n^2)$ for n line segments, which is too high for practical use in complex scenes.

One method for limiting the complexity of this process is to realize that proximity is an important variable in all of the significance measures. Since significance decreases with the square of separation, two small segments that are widely separated in the image are unlikely to produce significant groupings regardless of their other characteristics (constraints on measurement accuracy limit the contribution that orientation or other measurements can make to judging significance). Therefore, complexity can be limited by searching only a relatively small region surrounding each segment for candidates for grouping. Since proximity is judged relative to the size of the component features, the size of the region that must be searched is proportional to the length of the line segment from which we are initiating the search. In order to make efficient use of these restrictions, all segments in the image should be indexed in a grid-like data structure according to the position of each endpoint. For further efficiency, the segments in each element of this position matrix can be further indexed according to orientation and length. The use of this index allows all groupings with interesting levels of significance to be detected in time that is essentially linear in the number of features. It is

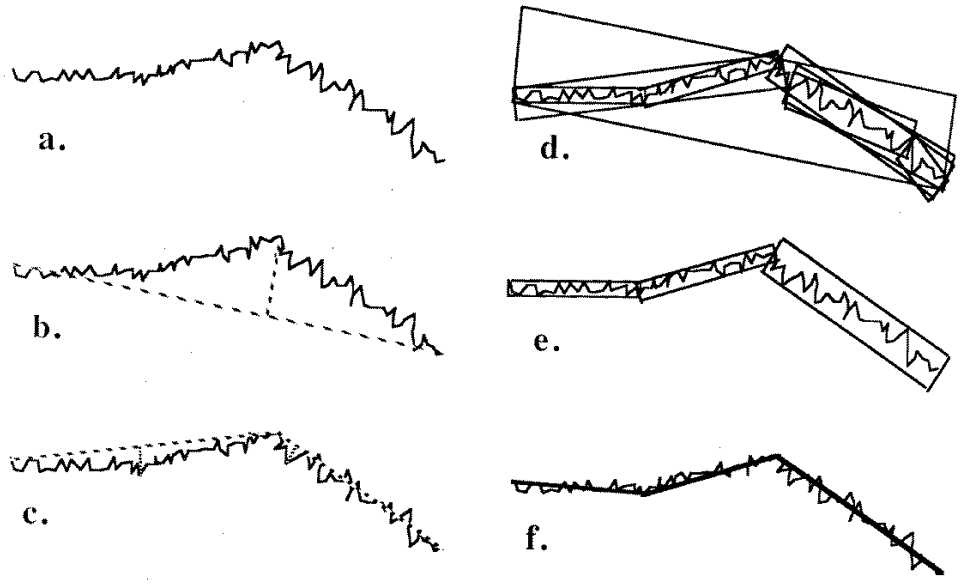

Figure 7: This illustrates the steps of a scale-independent algorithm for subdividing a curve into its most perceptually significant straight line segments. The input curve is shown in (a) and the final segmentation is given in (f).

interesting to note that human vision also seems to limit the complexity of grouping in a similar way, although human vision apparently uses a more sophisticated method that also takes account of the local density of features [19].

4.6 Segmentation of linked points into straight line segments

The examples above have dealt with the grouping of line segments. However, the derivation of the line segments themselves is a very important segmentation problem that is based upon detecting significant instances of collinearity among edge points. Most common edge detection methods produce linked lists of points as their output (e.g., points which lie on the zero-crossing of an image convolved with a second-derivative operator). In order to carry out the higher forms of perceptual organization described above, these linked points must be grouped into line or curve descriptions that make explict the significiant curvilinear structures at all scales. The author has previously described a method for finding straight-line and constant-curvature segmentations at multiple scales and for measuring their significance [19, Chap. 4]. However, here we will use a simplified method that selects only the single highest-significance line representation at each point along the curve.

The significance of a straight line fit to a list of points can be estimated by calculating the ratio of the length of the line segment divided by the maximum deviation of any point from the line (the maximum deviation is always assumed to be at least two pixels in size to account for limitations on measurement accuracy). This measure will remain constant as the scale of the image is changed, and it therefore provides a scale-

independent measure of significance that places no prior expectations on the allowable deviations. This significance measure is then used in a modified version of the recursive endpoint subdivision method (see Figure 7). A segment is recursively subdivided at the point with maximum deviation from a line connecting its endpoints (Figure 7 (b,c)). This process is repeated until each segment is no more than 4 pixels in length, producing a binary tree of possible subdivisions. This representation is similar to the strip trees described by Ballard [1]. Then, unwinding the recursion back up the tree, a decision is made at each junction as to whether to replace the current lower-level description with the single higher-level segment. The significance of every subsegment is calculated by its length-to-deviation ratio mentioned above (Figure 7 (d)). If the maximum significance of any of the subsegments is greater than the significance of the complete segment, then the subsegments are returned. Otherwise the single segment is returned. The procedure will return a segment covering every point along the curve (Figure 7 (e)). Finally, any segments with a length-to-deviation ratio less than 4 are discarded.

This algorithm is implemented in only 40 lines of Lisp code, yet it does a reasonable job of detecting the most perceptually significant straight line groupings in the linked point data. An advantage compared to the methods traditionally used in computer vision (which usually set some prior threshold for the amount of "noise" to be removed from a curve), is that it will tend to find the same structures regardless of the size at which an object appears in an image. In addition, it will avoid breaking up long lines if its shorter constituents do not appear to have a stronger perceptual basis.

5 The SCERPO vision system

The methods of spatial correspondence and perceptual organization described above have been combined to produce a functioning system for recognizing known three-dimensional objects in single gray-scale images. In order to produce a complete system, other components must also be included to perform low-level edge detection, object modeling, matching, and control functions. Figure 8 illustrates the various components and the sequence of information flow. Figures 9 to 16 show an example of the different stages of processing for an image of a randomly jumbled bin of disposable razors. Recognition was performed without any assumptions regarding orientation, position or scale of the objects; however, the focal length of the camera was specified in advance.

In order to provide the initial image features for input to the perceptual grouping process, the first few levels of image analysis in SCERPO use established methods of edge detection. The 512×512 pixel image shown in Figure 9 was digitized from the output of an inexpensive vidicon television camera. This image was convolved with a Laplacian of Gaussian function ($\sigma = 1.8$ pixels) as suggested by the Marr-Hildreth [23] theory of edge detection. Edges in the image should give rise to zero-crossings in this convolution, but where the intensity gradient is low there will also be many other zero-crossings that do not correspond to significant intensity changes in the image. Therefore, the Sobel gradient operator was used to measure the gradient of the image follow-

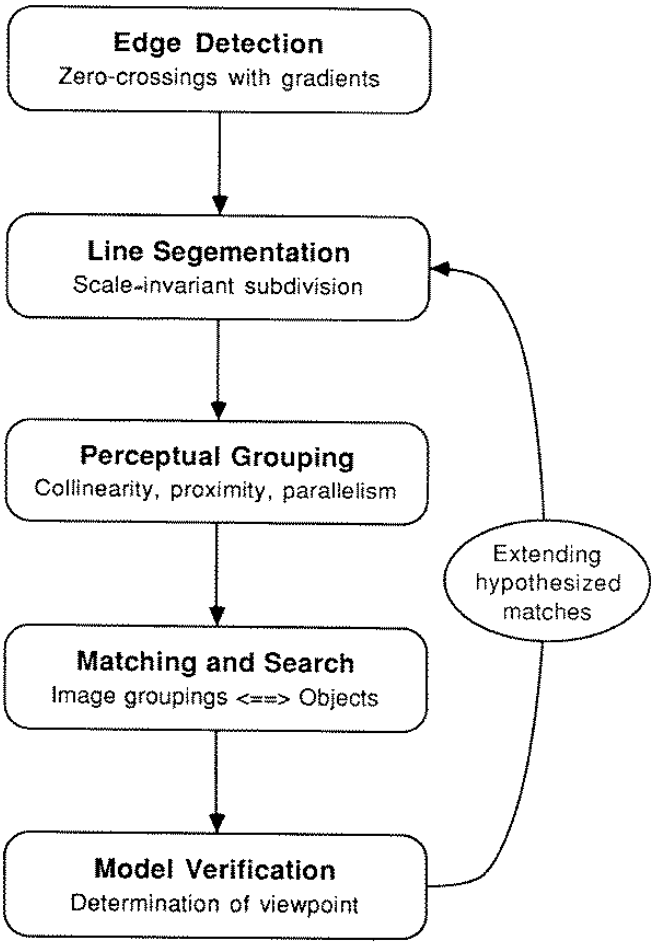

Figure 8: The components of the SCERPO vision system and the sequence of computation.

ing the $\nabla^2 G$ convolution, and this gradient value was retained for each point along a zero-crossing as an estimate of the signal-to-noise ratio. Figure 10 shows the resulting zero-crossings in which the brightness of each point along the zero-crossings is proportional to the magnitude of the gradient at that point.

These initial steps of processing were performed on a VICOM image processor under the *Vsh* software facility developed by Robert Hummel and Dayton Clark [7]. The VICOM can perform a 3×3 convolution against the entire image in a single video frame time. The *Vsh* software facility allowed the 18×18 convolution kernel required for our purposes to be automatically decomposed into 36 of the 3×3 primitive convolutions along with the appropriate image translations and additions. More efficient implementations which apply a smaller Laplacian convolution kernel to the image followed by iterated Gaussian blur were rejected due to their numerical imprecision. The VICOM is used only for the steps leading up to Figure 10, after which the zero-crossing image is transferred to a VAX 11/785 running UNIX 4.3 for subsequent processing. A program

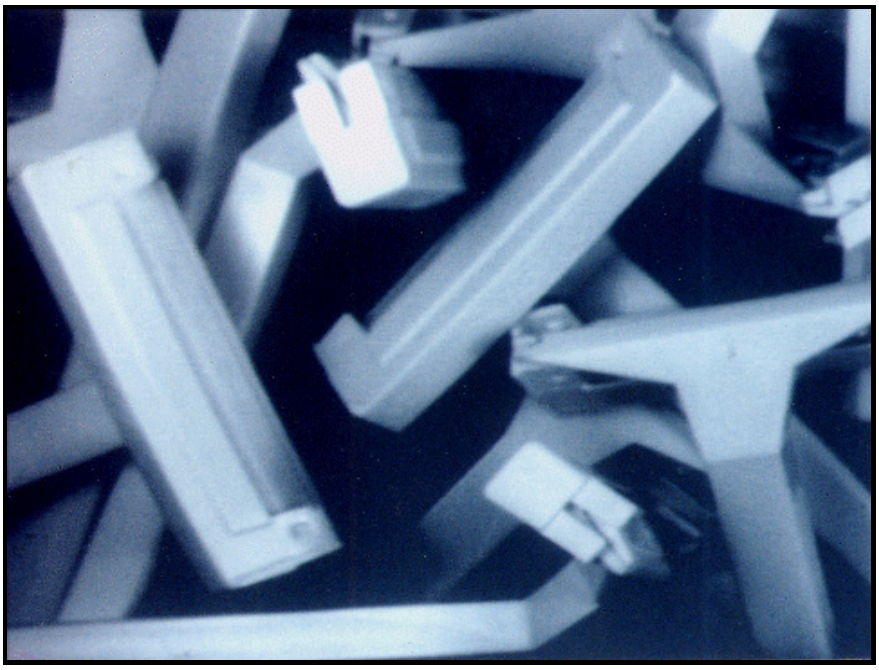

Figure 9: The original image of a bin of disposable razors, taken at a resolution of 512×512 pixels.

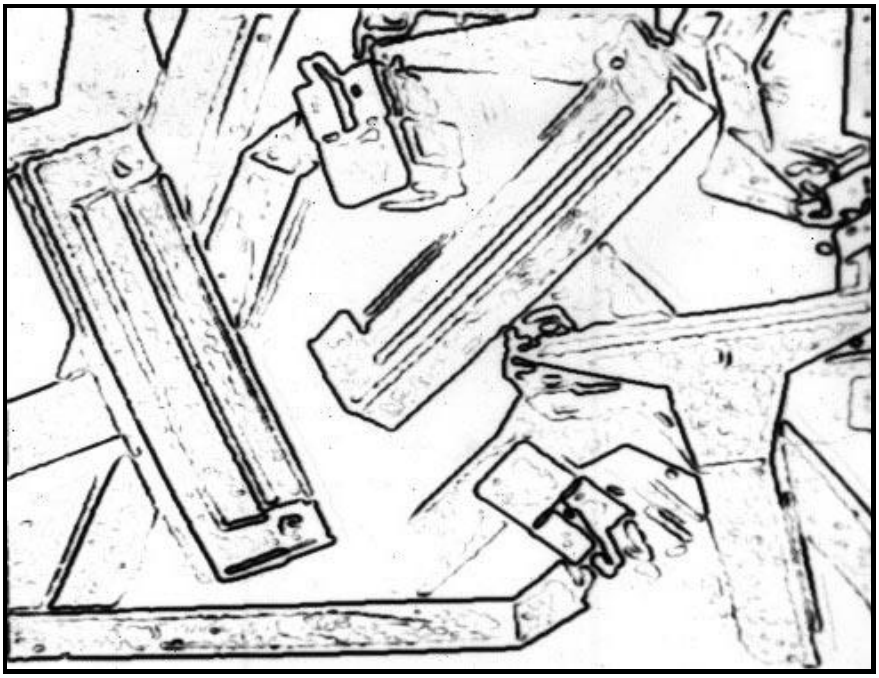

Figure 10: The zero-crossings of a $\nabla^2 G$ convolution. Grey levels are proportional to gradient magnitude at the zero-crossing.

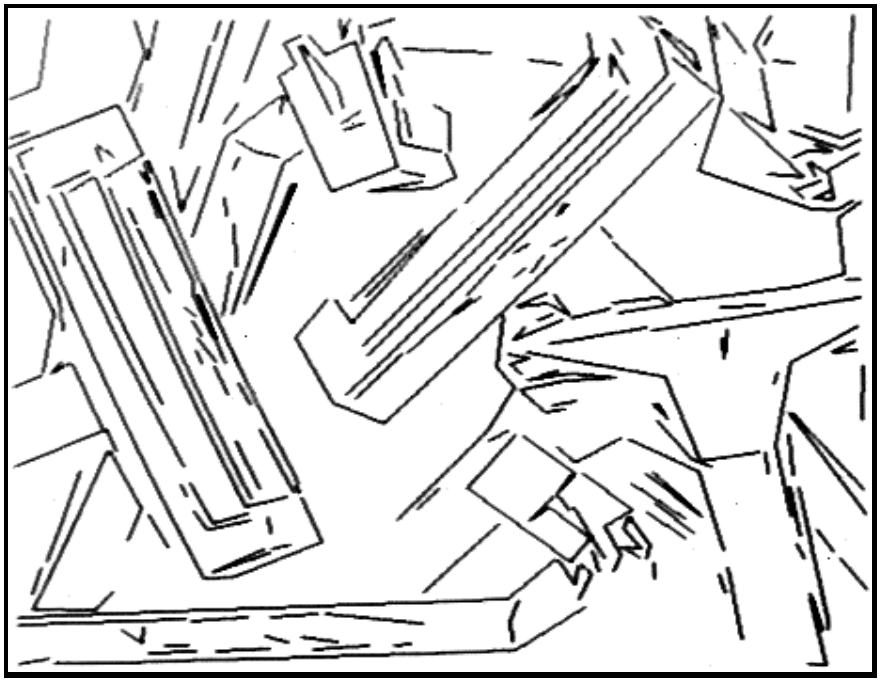

Figure 11: Straight line segments derived from the zero-crossing data of Figure 10 by a scale-independent segmentation algorithm.

written in C reads the zero-crossing image and produces a file of linked edge points along with the gradient magnitude at each point. All other components are written in Franz Lisp.

The next step of processing is to break the linked lists of zero-crossings into perceptually significant straight line segments, using the algorithm described in the previous section. Segments are retained only if the average gradient magnitude along their length is above a given threshold. It is much better to apply this threshold following segmentation than to the original zero-crossing image, since it prevents a long edge from being broken into shorter segments when small portions dip below the gradient threshold. The results of performing these operations are shown in Figure 11. The recognition problem now consists of searching among this set of about 300 straight line segments for subsets that are each spatially consistent with model edges projected from a single viewpoint.

The straight line segments are indexed according to endpoint locations and orientation. Then a sequence of procedures is executed to detect significant instances of collinearity, endpoint proximity (connectivity), and parallelism. Each instance of these relations is assigned a level of significance using the formulas given above in the section on perceptual organization. Pointers are maintained from each image segment to each of the other segments with which it forms a significant grouping. These primitive relations could be matched directly against corresponding structures on the three-dimensional object models, but the search space for this matching would be large due to the substantial remaining level of ambiguity. The size of the search space can be re-

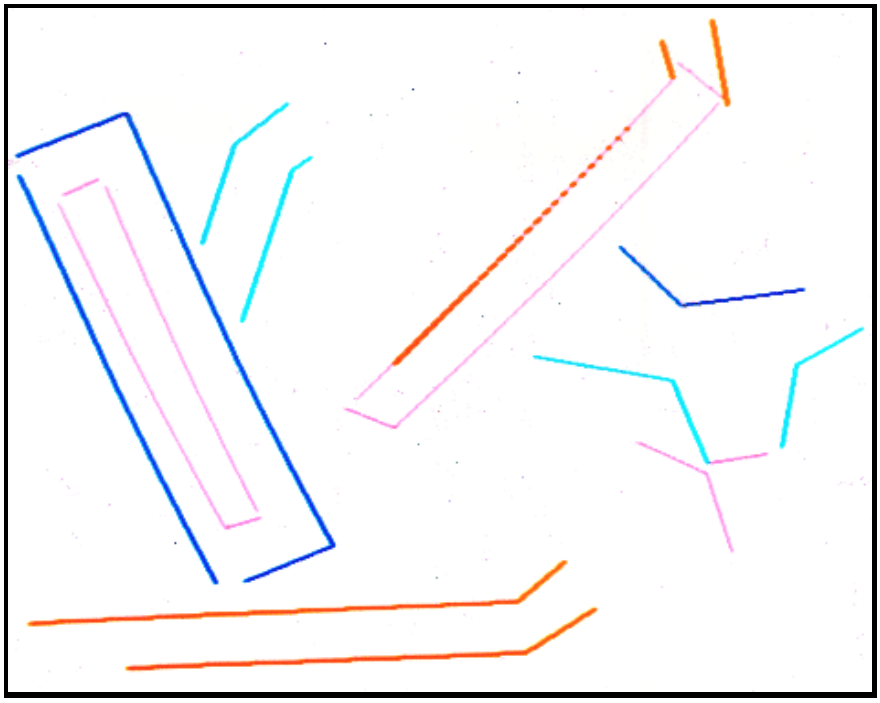

Figure 12: The most highly-ranked perceptual groupings detected from among the set of line segments. There is some overlap between groupings.

duced by first combining the primitive relations into larger, more complex structures.

The larger structures are found by searching among the graph of primitive relations for specific combinations of relations which share some of the same line segments. For example, trapezoid shapes are detected by examining each pair of parallel segments for proximity relations to other segments which have both endpoints in close proximity to the endpoints of the two parallel segments. Parallel segments are also examined to detect other segments with proximity relations to their endpoints which were themselves parallel. Another higher-level grouping is formed by checking pairs of proximity relations that are close to one another to see whether the four segments satisfy Kanade's [16] skewed symmetry relation (i.e., whether the segments could be the projection of segments that are bilaterally symmetric in three-space). Since each of these compound structures is built from primitive relations that are themselves viewpoint-invariant, the larger groupings also reflect properties of a three-dimensional object that are invariant over a wide range of viewpoints. The significance value for each of the larger structures is calculated by simply multiplying together the probabilities of non-accidentalness for each component. These values are used to rank all of the groupings in order of decreasing significance, so that the search can begin with those groupings that are most perceptually significant and are least likely to have arisen through some accident. Figure 12 shows a number of the most highly-ranked groupings that were detected among the segments of Figure 11. Each of the higher-level groupings in this figure contains four line segments, but there is some overlap in which a single line segment partici-

pates in more than one of the high-level groupings.

Even after this higher-level grouping process, the SCERPO system clearly makes use of simpler groupings than would be needed by a system that contained large numbers of object models. When only a few models are being considered for matching, it is possible to use simple groupings because even with the resulting ambiguity there are only a relatively small number of potential matches to examine. However, with large numbers of object models, it would be necessary to find more complex viewpoint-invariant structures that could be used to index into the database of models. The best approach would be to make use of some form of evidential reasoning to combine probabilistic information from multiple sources to limit the size of the search space. This approach has been outlined by the author in earlier work [19, Chap. 6].

5.1 Model matching

The matching process consists of individually comparing each of the perceptual groupings in the image against each of the structures of the object model which is likely to give rise to that form of grouping. For each of these matches, the verification procedure is executed to solve for viewpoint, extend the match, and return an answer as to whether the original match was correct. Given the large number of potential matches and their varied potential for success, it is important to make use of a ranking method to select the most promising matches first. For example, every straight line detected in the image is a form of grouping that could be matched against every straight edge of the model, but this would involve a large amount of search. In general, the more complex a grouping is the fewer potential matches it will have against the model. Therefore, the only matches that are considered in the current implementation are those that involve image groupings that contain at least 3 line segments. This also has the important result that such a grouping will generally contain enough information to solve exactly for viewpoint from the initial match, which provides tight constraints to speed up the operation of the verification procedure.

A more precise specification of the optimal ordering for the search process could be stated as follows. In order to minimize the search time, we would like to order our consideration of hypotheses according to decreasing values of P_k/W_k, where P_k is the probability that a particular hypothesis for the presence of object k is correct, and W_k is the amount of work required to verify or refute it. In general, increased complexity of a grouping will lead to fewer potential matches against different object features, and therefore will increase the probability P_k that any particular match is correct. However, this probability is also very dependent upon the particular set of objects that are being considered, since some features will function more effectively to discriminate among one particular set of objects than in another set. The most effective way to determine the optimal values of P_k for each potential match would be through the use of a learning procedure in which the actual probability values are refined through experience in performing the recognition task. These probability adjustments would in essence be strengthening or weakening associations between particular image features and partic-

ular objects. These object-specific values would be multiplied by the probability that a grouping is non-accidental to determine the final estimate of P_k. The W_k could be similarly learned through experience. The current implementation of SCERPO simply uses the complexity of a grouping as a crude estimate for these desired ranking parameters.

In order to speed the runtime performance of the matching process, the viewpoint-invariant groupings that each model can produce in the image are precomputed off-line. The model is simply checked for three-dimensional instances of the three primitive image relations that are detected during the perceptual grouping process: i.e., connectivity, collinearity, and parallelism. No attempt is made to find approximate instances of these relations, so in essence the relations are implicitly specified by the user during model input. These relations are then grouped into the same types of larger structures that are created during the perceptual grouping process, and are stored in separate lists according to the type of the grouping. Any rotational symmetries or other ambiguities are used to create new elements in the lists of possible matches. The runtime matching process therefore consists only of matching each image grouping against each element of a precomputed list of model groupings of the same type.

One important feature of this matching process is that it is opportunistic in its ability to find and use the most useful groupings in any particular image. Given that there is no prior knowledge of specific occlusions, viewpoint, amount of noise, or failures in the detection process, it is important to use a run-time ranking process that selects among the actual groupings that are found in any particular image in order to make use of those that are most likely to be non-accidental. Since each view of an object is likely to give rise to many characteristic perceptual groupings in an image, it is usually possible to find features to initiate the search process even when a substantial portion of the object is occluded. The current implementation of SCERPO is at a rather simple level in terms of the sophistication of perceptual grouping and matching, and as should be clear from the above discussion there are many opportunities for making these processes faster and more complete. As more types of groupings and other viewpoint-invariant features are added, the number of possible matches would increase, but the expected amount of search required for a successful match would decrease due to the increased specificity of the matches with the highest rankings.

The implementation of the viewpoint-solving and verification process has already been described in detail in an earlier section. This is the most robust and reliable component of the system, and its high level of performance in extending and verifying a match can compensate for many weaknesses at the earlier stages. The low probability of false positives in this component means that failure at the earlier levels tends to result simply in an increased search space rather than incorrect matches. Figure 13 shows the three-dimensional model that was used for matching against the image data. Figure 14 shows the final set of successful matches between particular viewpoints of this model and sets of image segments. Once a particular instance of an object has been identified, the matched image segments are marked to indicate that they may no longer participate in any further matches. Any groupings which contain one of these marked segments are ignored during the continuation of the matching process. Therefore, as instances of

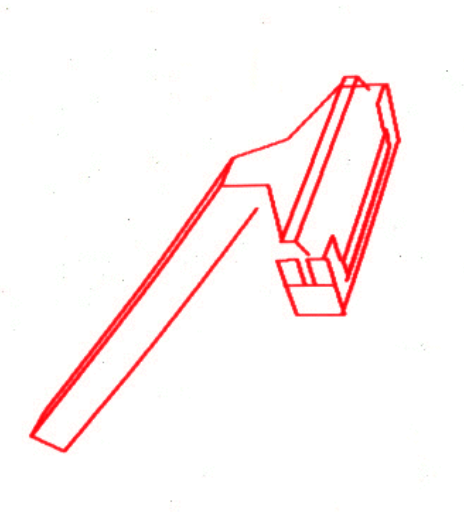

Figure 13: The three-dimensional wire-frame model of the razor shown from a single viewpoint.

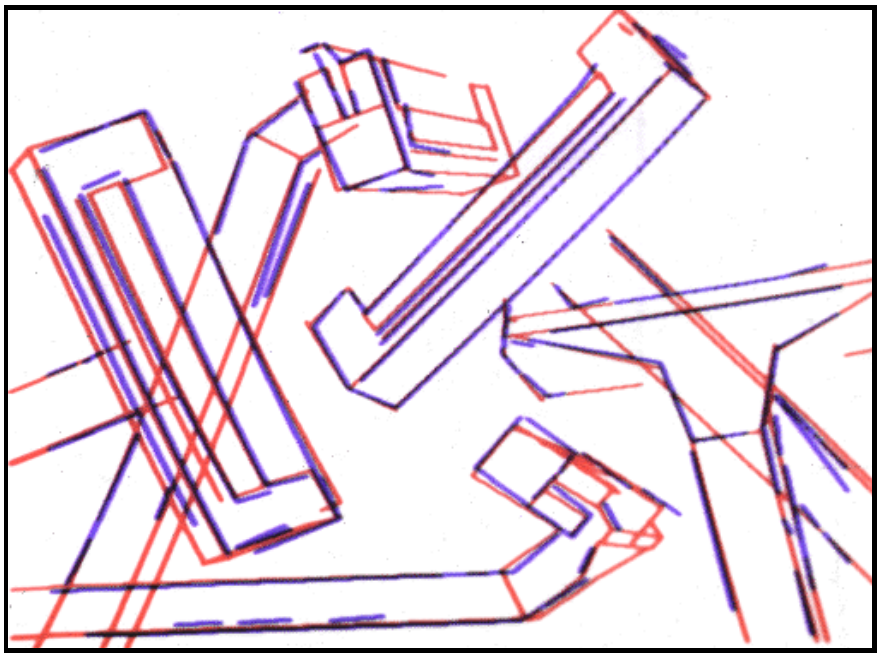

Figure 14: Successful matches between sets of image segments and particular viewpoints of the model.

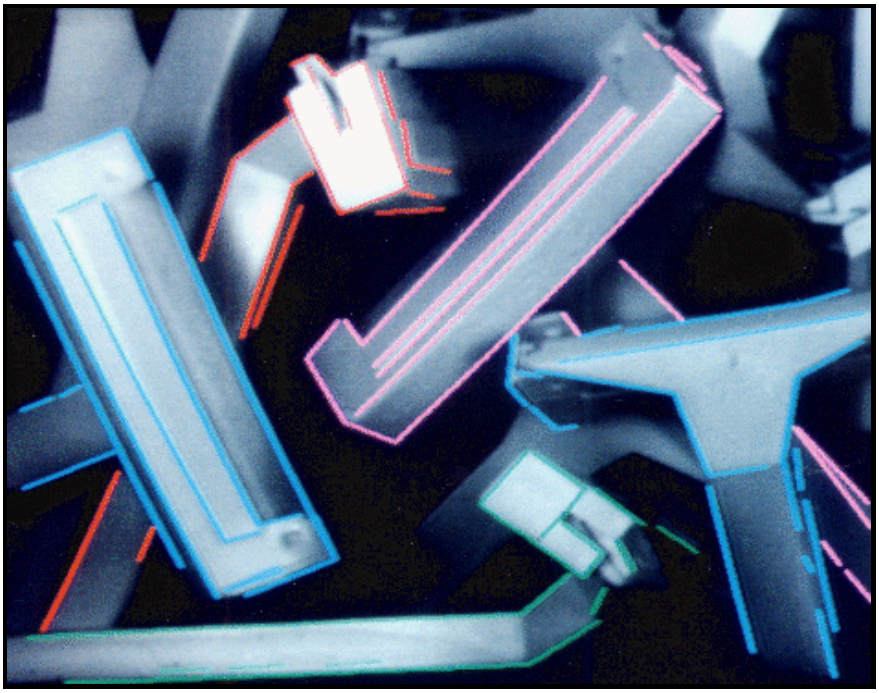

Figure 15: Successfully matched image segments superimposed upon the original image.

the object are recognized, the search space can actually decrease for recognition of the remaining objects in the image.

Since the final viewpoint estimate is performed by a least-squares fit to greatly over-constrained data, its accuracy can be quite high. Figure 15 shows the set of successfully matched image segments superimposed upon the original image. Figure 16 shows the model projected from the final calculated viewpoints, also shown superimposed upon the original image. The model edges in this image are drawn with solid lines where there is a matching image segment and with dotted lines over intervals where no corresponding image segment could be found. The accuracy of the final viewpoint estimates could be improved by returning to the original zero-crossing data or even the original image for accurate measurement of particular edge locations. However, the existing accuracy should be more than adequate for typical tasks involving mechanical manipulation.

The current implementation of SCERPO is designed as a research and demonstration project, and much further work would be required to develop the speed and generality needed for many applications. Relatively little effort has been devoted to minimizing computation time. The image processing components require only a few seconds of computation on the VICOM image processor, but then the image must be transferred to a VAX 11/785 running UNIX 4.3 for further processing. It requires about 20 seconds for a program written in C to read the zero-crossing image and output a file of linked edge points. This file is read by a Franz Lisp routine, and all subsequent processing

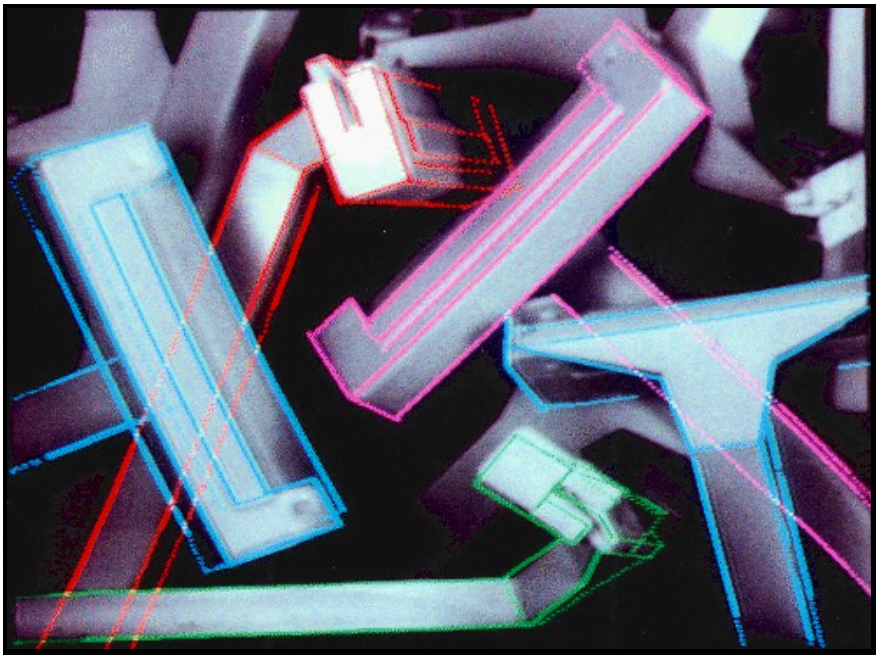

Figure 16: The model projected onto the image from the final calculated viewpoints. Model edges are shown dotted where there was no match to a corresponding image segment.

is performed within the Franz Lisp virtual memory environment. Segmentation into straight lines requires 25 seconds, indexing and grouping operations require about 90 seconds and the later stages of matching and verification took 65 seconds for this example. There are numerous ways in which the code could be improved to reduce the required amount of computation time if this were a major goal. Careful design of data structures would allow fast access of image segments according to predicted positions, lengths, and orientations. Each iteration of the crucial viewpoint-solving process requires at most several hundred floating point operations, so there is reason to believe that a carefully coded version of the basic search loop could run at high rates of speed.

6 Directions for future research

The most obvious direction in which to extend the current system is to generalize the object models to include many new types of visual knowledge. These extensions could include the modeling of moveable articulations, optional components, and other variable parameters in the models. The section above on solving for spatial correspondence described methods for incorporating these extensions during the viewpoint-solving and matching process. However, further research is required to determine the optimal order in which to solve for individual parameters. Imagine, for example, that we had a generic model of the human face. The model would include small ranges of variation for the size and position of every feature, as well as many optional components such

as a beard or glasses. However, given some tentative correspondences for say, the eyes and nose, we could use the expectation of bilateral symmetry and the most tightly constrained dimensions of our model to solve for approximate viewpoint. This would then suggest quite tightly constrained regions in which to search for other features, such as ears, chin, eyebrows, etc., each of which could be used to derive better estimates of viewpoint and the other parameters. The resulting values of these parameters could then be mapped into a feature space and used to identify particular individuals, which in turn may lead to further detailed constraints and expectations. Some mechanisms for ordering these constraints were incorporated into the ACRONYM system [6].

The current implementation of SCERPO has used only an edge-based description of the image because that is a comparatively reliable and well-researched form of image analysis. But the same framework could incorporate many other dimensions of comparison between model and image, including areas such as surface modeling, texture, color, and shading properties. Further research would be required to detect viewpoint-invariant aspects of these properties during bottom-up image analysis. Many of the modeling and predictive aspects of these problems have been developed for use in computer graphics, but it may be necessary to find faster ways to perform these computations for use in a computer vision system.

Once a number of different sources of information are used to achieve the optimal ordering of the search process, it is necessary to make use of general methods for combining multiple sources of evidence. The use of evidential reasoning for this problem has been discussed elsewhere by the author in some detail [19, Chap. 6]. These methods make use of prior estimates for the probability of the presence of each object, and then update these estimates as each new source of evidence is uncovered. Sources of evidence might include particular perceptual groupings, colors, textures, and contextual information. Context plays an important role in general vision, since most scenes will contain some easily-identified objects which then provide a great deal of information regarding size, location, and environment which can greatly ease the recognition of more difficult components of the scene. Evidential reasoning also provides an opportunity to incorporate a significant form of learning, since the conditional probability estimates can be continuously adjusted towards their optimal values as a system gains experience with its visual environment. In this way associations could be automatically created between particular objects and the viewpoint-invariant features to which they are likely to give rise in the image.

The psychological implications of this research are also deserving of further study. Presumably human vision does not perform a serial search of the type used in the SCERPO system. Instead, the brief time required for typical instances of recognition indicates that any search over a range of possible objects and parameters must be occurring in parallel. Yet even the human brain does not contain enough computational power to search over every possible object at every viewpoint and position in the image. This can be demonstrated by the fact that even vague non-visual contextual clues can decrease the length of time required to recognize degraded images [17]. Presumably, if a complete search were being performed in every instance, any top-down clues that narrowed

the search would have little effect. Given that the search is proceeding in parallel, the mechanisms used for ranking the search in SCERPO would instead be used to select a number of possibilities to explore in parallel, limited according to the available computational resources. This model for the recognition process suggests many psychophysical experiments in which average recognition times could be measured for different combinations of image data and contextual information. Some important experimental results relating recognition time to the availability of various image and contextual clues have been reported by Biederman [4].

7 Related research on model-based vision

The methods used in the SCERPO system are based on a considerable body of previous research in model-based vision. The pathbreaking early work of Roberts [24] demonstrated the recognition of certain polyhedral objects by exactly solving for viewpoint and object parameters. Matching was performed by searching for correspondences between junctions found in the scene and junctions of model edges. Verification was then based upon exact solution of viewpoint and model parameters using a method that required seven point-to-point correspondences. Unfortunately, this work was poorly incorporated into later vision research, which instead tended to emphasize non-quantitative and much less robust methods such as line-labeling.

The ACRONYM system of Brooks [6] used a general symbolic constraint solver to calculate bounds on viewpoint and model parameters from image measurements. Matching was performed by looking for particular sizes of elongated structures in the image (known as ribbons) and matching them to potentially corresponding parts of the model. The bounds given by the constraint solver were then used to check the consistency of all potential matches of ribbons to object components. While providing an influential and very general framework, the actual calculation of bounds for such general constraints was mathematically difficult and approximations had to be used that did not lead to exact solutions for viewpoint. In practice, prior bounds on viewpoint were required which prevented application of the system to full three-dimensional ranges of viewpoints.

Goad [12] has described the use of automatic programming methods to precompute a highly efficient search path and viewpoint-solving technique for each object to be recognized. Recognition is performed largely through exhaustive search, but precomputation of selected parameter ranges allows each match to place tight viewpoint constraints on the possible locations of further matches. Although the search tree is broad at the highest levels, after about 3 levels of matching the viewpoint is essentially constrained to a single position and little further search is required. The precomputation not only allows the fast computation of the viewpoint constraints at runtime, but it also can be used at the lowest levels to perform edge-detection only within the predicted bounds and at the minimum required resolution. This research has been incorporated in an industrial computer vision system by Silma Inc. which has the remarkable capa-

bility of performing all aspects of three-dimensional object recognition within as little as 1 second on a single microprocessor. Because of their extreme runtime efficiency, these precomputation techniques are likely to remain the method of choice for industrial systems dealing with small numbers of objects.

Other closely related research on model-based vision has been performed by Shirai [26] and Walter & Tropf [28]. There has also been a substantial amount of research on the interpretation of range data and matching within the three-dimensional domain. While we have argued here that most instances of recognition can be performed without the preliminary reconstruction of depth, there may be industrial applications in which the measurement of many precise three-dimensional coordinates is of sufficient importance to require the use of a scanning depth sensor. Grimson & Lozano-Pérez [13] have described the use of three-dimensional search techniques to recognize objects from range data, and describe how these same methods could be used with tactile data, which naturally occurs in three-dimensional form. Further significant research on recognition from range data has been carried out by Bolles *et al.* [5] and Faugeras [10]. Schwartz & Sharir [25] have described a fast algorithm for finding an optimal least-squares match between arbitrary curve segments in two or three dimensions. This method has been combined with the efficient indexing of models to demonstrate the recognition of large numbers of two-dimensional models from their partially obscured silhouettes. This method also shows much promise for extension to the three-dimensional domain using range data.

8 Conclusions

One goal of this paper has been to describe the implementation of a particular computer vision system. However, a more important objective for the long-term development of this line of research has been to present a general framework for attacking the problem of visual recognition. This framework does not rely upon any attempt to derive depth measurements bottom-up from the image, although this information could be used if it were available. Instead, the bottom-up description of an image is aimed at producing viewpoint-invariant groupings of image features that can be judged unlikely to be accidental in origin even in the absence of specific information regarding which objects may be present. These groupings are not used for final identification of objects, but rather serve as "trigger features" to reduce the amount of search that would otherwise be required. Actual identification is based upon the full use of the viewpoint consistency constraint, and maps the object-level data right back to the image level without any need for the intervening grouping constructs. This interplay between viewpoint-invariant analysis for bottom-up processing and viewpoint-dependent analysis for top-down processing provides the best of both worlds in terms of generality and accurate identification. Many other computer vision systems have experienced difficulties because they attempt to use viewpoint-specific features early in the recognition process or because they attempt to identify an object simply on the basis of viewpoint-invariant

characteristics. The many quantitative constraints generated by the viewpoint consistency analysis allow for robust performance even in the presence of only partial image data, which is one of the most basic hallmarks of human vision.

There has been a tendency in computer vision to concentrate on the low-level aspects of vision because it is presumed that good data at this level is prerequisite to reasonable performance at the higher levels. However, without any widely accepted framework for the higher levels, the development of the low level components is proceeding in a vacuum without an explicit measure for what would constitute success. This situation encourages the idea that the purpose of low-level vision should be to recover explict physical properties of the scene, since this goal can at least be judged in its own terms. But recognition does not depend on physical properties so much as on stable *visual* properties. This is necessary so that recognition can occur even in the absence of the extensive information that would be required for the bottom-up physical reconstruction of the scene. If a widely accepted framework could be developed for high-level visual recognition, then it would provide a whole new set of criteria for evaluating work at the lower levels. We have suggested examples of such criteria in terms of viewpoint invariance and the ability to distinguish significant features from accidental instances. If such a framework were adopted, then rapid advances could be made in recognition capabilities by independent research efforts to incorporate many new forms of visual information.

9 Acknowledgments

This research was supported by NSF grant DCR-8502009. Implementation of the SCERPO system relied upon the extensive facilities and software of the NYU vision laboratory, which are due to the efforts of Robert Hummel, Jack Schwartz, and many others. Robert Hummel, in particular, provided many important kinds of technical and practical assistance during the implementation process. Mike Overton provided help with the numerical aspects of the design. Much of the theoretical basis for this research was developed while the author was at the Stanford Artificial Intelligence Laboratory, with the help of Tom Binford, Rod Brooks, Chris Goad, David Marimont, Andy Witkin, and many others.

References

[1] Ballard, D.H., "Strip trees: a hierarchical representation for curves," *Communications of the ACM,* **24**, 5 (May 1981), 310-321.

[2] Barnard, Stephen T., "Interpreting perspective images," *Artificial Intelligence,* **21** (1983), 435-462.

[3] Barrow, H.G. and J.M. Tenenbaum, "Interpreting line drawings as three-dimensional surfaces," *Artificial Intelligence,* **17** (1981), 75-116.

[4] Biederman, Irving, "Human image understanding: recent research and a theory," *Computer Vision, Graphics, and Image Processing,* **32** (1985), 29-73.

[5] Bolles, R.C., P. Horaud, and M.J. Hannah, "3DPO: A three-dimensional part orientation system," *Proc. of 8th International Joint Conf. on Artificial Intelligence* (Karlsruhe, West Germany, 1983), 1116-1120.

[6] Brooks, Rodney A., "Symbolic reasoning among 3-D models and 2-D images," *Artificial Intelligence,* **17** (1981), 285-348.

[7] Clark, Dayton and Robert Hummel, "VSH user's manual: an image processing environment," *Robotics Research Technical Report,* Courant Institute, New York University (September 1984).

[8] Conte, S.D. and Carl de Boor, *Elementary Numerical Analysis: An Algorithmic Approach, Third Edition* (New York: McGraw-Hill, 1980).

[9] Fischler, Martin A. and Robert C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM,* **24,** 6 (1981), 381-395.

[10] Faugeras, O.D., "New steps toward a flexible 3-D vision system for robotics," *Proc. of 7th International Conference on Pattern Recognition* (Montreal, 1984), 796-805.

[11] Gibson, J.J., *The Ecological Approach to Visual Perception* (Boston: Houghton Mifflin, 1979).

[12] Goad, Chris, "Special purpose automatic programming for 3D model-based vision," *Proceedings ARPA Image Understanding Workshop,* Arlington, Virginia (1983).

[13] Grimson, Eric, and Thomás Lozano-Pérez, "Model-based recognition and localization from sparse range or tactile data," *Int. Journal of Robotics Research,* **3** (1984), 3-35.

[14] Hochberg, Julian E., "Effects of the Gestalt revolution: The Cornell symposium on perception," *Psychological Review,* **64,** 2 (1957), 73-84.

[15] Hochberg, Julian E. and Virginia Brooks, "Pictorial recognition as an unlearned ability: A study of one child's performance," *American Journal of Psychology,* **75** (1962), 624-628.

[16] Kanade, Takeo, "Recovery of the three-dimensional shape of an object from a single view," *Artificial Intelligence,* **17** (1981), 409-460.

[17] Leeper, Robert, "A study of a neglected portion of the field of learning—the development of sensory organization," *Journal of Genetic Psychology,* **46** (1935), 41-75.

[18] Lowe, David G., "Solving for the parameters of object models from image descriptions," *Proc. ARPA Image Understanding Workshop* (College Park, MD, April 1980), 121–127.

[19] Lowe, David G., *Perceptual Organization and Visual Recognition* (Boston, Mass: Kluwer Academic Publishers, 1985).

[20] Lowe, David G., and Thomas O. Binford, "The recovery of three-dimensional structure from image curves," *IEEE Trans. on Pattern Analysis and Machine Intelligence,* **7,** 3 (May 1985), 320-326.

[21] Marr, David, "Early processing of visual information," *Philosophical Transactions of the Royal Society of London, Series B,* **275** (1976), 483-524.

[22] Marr, David, *Vision* (San Francisco: W.H. Freeman and Co., 1982).

[23] Marr, David, and Ellen Hildreth, "Theory of edge detection," *Proc. Royal Society of London, B,* **207** (1980), 187-217.

[24] Roberts, L.G., "Machine perception of three-dimensional solids," in *Optical and Electro-optical Information Processing,* J. Tippet, Ed. (Cambridge, Mass.: MIT Press, 1966), 159-197.

[25] Schwartz, J.T. and M. Sharir, "Identification of partially obscured objects in two dimensions by matching of noisy characteristic curves," *Tech. Report 165, Courant Institute, New York University* (June 1985).

[26] Shirai, Y., "Recognition of man-made objects using edge cues," in *Computer Vision Systems,* A. Hanson, E. Riseman, eds. (New York: Academic Press, 1978).

[27] Stevens, Kent A., "The visual interpretation of surface contours," *Artificial Intelligence,* **17** (1981), 47-73.

[28] Walter, I. and H. Tropf, "3-D recognition of randomly oriented parts," *Proceedings of the Third International Conf. on Robot Vision and Sensory Controls* (November, 1983, Cambridge, Mass.), 193–200.

[29] Wertheimer, Max, "Untersuchungen zur Lehe von der Gestalt II," *Psychol. Forsch.,* **4** (1923). Translated as "Principles of perceptual organization" in *Readings in Perception,* David Beardslee and Michael Wertheimer, Eds., (Princeton, N.J.: Van Nostrand, 1958), 115-135.

[30] Witkin, Andrew P. and Jay M. Tenenbaum, "On the role of structure in vision," in *Human and Machine Vision,* Beck, Hope & Rosenfeld, Eds. (New York: Academic Press, 1983), 481-543.

[31] Wolf, Paul R., *Elements of Photogrammetry* (New York: McGraw-Hill, 1983).

[32] Zucker, Steven W., "Computational and psychophysical experiments in grouping: Early orientation selection," in *Human and Machine Vision,* Beck, Hope & Rosenfeld, Eds. (New York: Academic Press, 1983), 545-567.