Perceptual Organization for Scene Segmentation and Description

Rakesh Mohan, *Member, IEEE*, and Ramakant Nevatia, *Fellow, IEEE*

***Abstract*— A novel data-driven system for segmenting scenes into objects and their components is presented. This segmentation system generates hierarchies of features that correspond to structural elements such as boundaries and surfaces of objects. The technique is based on perceptual organization. In humans, perceptual organization is the ability to readily group elements in an image based on various realtionships between them. Here, perceptual organization is implemented as a mechanism to exploit geometrical regularities in the shapes of objects as projected onto images.**

Edges are recursively grouped on geometrical relationships into a description hierarchy ranging from edges to the visible surfaces of objects. These edge groupings, which are termed collated features, are abstract descriptors encoding structural information. The geometrical relationships employed are quasiinvariant over 2-D projections and are common to structures of most objects. Thus, collations have a high likelihood of corresponding to parts of objects. Collations serve as intermediate and high-level features for various visual processes. Applications of collations to stereo correspondence, object level semgentation, and shape description are illustrated.

***Index Terms*—Constraint satisfaction networks, perceptual organization, segmentation, symmetry.**

I. Introduction

In the bottom-up interpretation of images with no recourse to explicit object models, vision systems exploit generic constraints on the physical world and the imaging process. Some constraints are similarity of intensity and color, smoothness [46] and rigidity. We explore geometric constraints, namely, certain geometric regularities are exhibited by the structure of most objects.

Segmentation systems use these bottom-up constraints to extract relevant features from images. For example, region segmentation partitions images into regions of uniform color or texture [40]. This paper addresses the issuse of segmenting images on structural rather than spectral or textural properties. Geometrical entities, such as structure and shape play a crucial role in vision [72], [49]. Indoor scenes and manmade objects are rich in organized structure. Thus, the presented system can augment traditional segmentation systems to handle scenes that are rich in structural information.

Manuscript received June 4, 1990; revised September 17, 1991. Recommended for acceptance by Associate Editor C. Brown.

R. Mohan is with the Exploratory Computer Vision Group, IBM Thomas J. Watson Research Center, Yorktown Heights, NY 10598.

R. Nevatia is with the Institute for Robotics and Intelligent Systems, University of Southern California, Los Angeles, CA 90089-0273.

IEEE Log Number 9107552.

Segmentation is critical in computer vision as it provides input to a wide variety of visual processes ranging from stereo and motion correspondence to model recognition. The performance of these processes is dependent on the quality of input features. For example, for model matching, Grimson [24] shows that without segmentation (i.e., the selection of edges or lines which correspond to the same object), recognition is exponential, but with even partial segmentation (most, but not all, of the features selected correspond to the same object), recognition time is polynomial.

Humans can immediately detect relationships such as collinearity, parallelism, connectivity, and repetitive patterns among image elements [43]. This phenomenon is called perceptual organization. Consider Fig. 1, where the image elements are organized into various structures on the basis of the geometrical relationships between the elements. This ability to group entities, based on some realtionship between them, is pervasive in various perceptual mechanisms (see [39] for some examples). We propose[1] that perceptual organization takes primitive image elements, which are typically generated by low-level segmentation processes, and generates representations of feature groupings that encode the structural interrelationships between the component elements. These representations, which are termed collated features, are used by various visual processes. Collated features are organized as description hierarchies representing much of the structural information in an image.

We concentrate on geometrical relationships in real images (for example, Fig. 2). We will not consider grouping on a scale smaller than that of the objects (for example, texture) or on the similarity in constrast and color. We will also not consider perceptual organization in the context of 3-D data or motion. First, we disucuss the broad issues in perceptual organization for computer vision (Sections II–IV). Next, we present a vision system CANC2 [48], which is based on perceptual organization (Sections V–XIV). It is designed to handle scenes of unkown curved objects imaged from arbitrary viewpoints. It generates a hierarchy of collated features by grouping edges recursively. The collations are potentially useful for a wide variety of visual tasks. Currently, the system uses these collations to segment intensity images into objects and their surfaces, generate 2-D shape descriptions of the projected surfaces, and perfrom stereo correspondence at the level of surfaces.

[1] While this work is based on ideas from perceptual organization in humans, it does not claim to model the human visual mechanism.

0162-8828/92$03.00

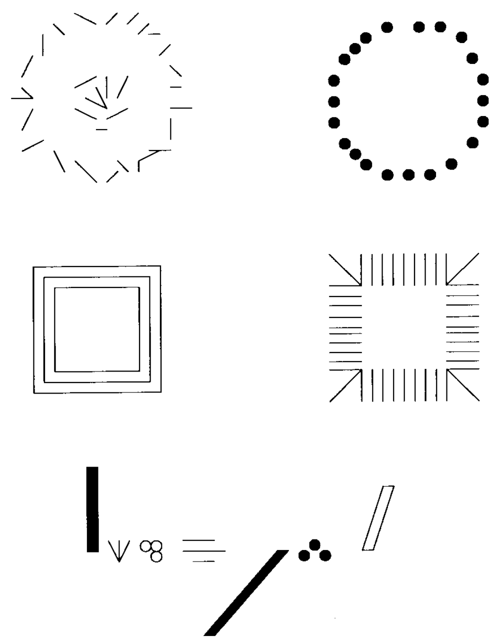


Fig. 1. Grouping on geometrical relationships (from [46]).

Fig. 2. Image I.

II. Related Work

Perceptual organization has been studied by investigators in psychology [2], [32], [36], [56], [66] and computer vision [43], [44], [72], [74], [37], [64], [19]. Excellent surveys of this work can be found in [43] and [56].

The human visual system exhibits a preference for some shapes and arrangements over others. What characterizes these shapes or arrangements? The Gestalt psychologists proposed the principle of Pragnanz or simplicity of form [70] but did not formalize it in computational terms. Later, it was quantified in terms of information theory by proposals that equated minimal description length to simplicity of form [2], [25], [41].

Garner and Clement [20], [21] explained observed goodness ratings for simple geometrical patterns in terms of their invariance to certain transformations. Palmer [56] develops their work further. He proposes that perceptual organization detects simplest organizations such as those that are most locally invariant over the group of Euclidian similarity transforms.

Marr [46] views perceptual organization as grouping processes that operate on the raw primal sketch (primitive descriptions of the image) to build up descriptive primitives. Witkin and Tenenbaum [72] present the concept of nonaccidentalness. Lowe extended the proposal of Witkin and Tenenbaum by computing prior probabilities of accidental and nonaccidental instances of certain relations (collinearity, proximity of end points, and parallelism) and using these to constrain the search in model matching [43].

In computer vision, a common application of perceptual organization has been to detect straight lines [6] and curves [14]. Reynolds and Beveridge [60] detect groupings of parallel lines and proximate orthogonal lines (among other relationships) in aerial images. These groupings indicate organization at the scene level and do not correspond to individual objects and, therefore, are not useful for segmenting individual objects. Quan *et al.* [58] have successfully used scene-level groupings to detect and estimate motion.

The authors have previously implemented a vision system [51] that uses perceptual organization to detect objects with particular shapes but without explicit models. The input was aerial images of urban areas and the output 3-D models of the buildings in the scene. The building roofs were assumed to be of a specific shape, namely, a combination of rectangles. For specific shapes, the relevant geometrical relationships for grouping can be obtained by decomposing the specific component shapes. One possible way of handling generic shapes and viewpoints is by extending the catalog of basic shapes (i.e., in addition to rectangles, consider shapes such as ellipses, polygons, etc.). This paper presents an alternate approach.

III. Perceptual Organization

In this section, three basic results in perceptual organization that have arisen from previous research (Section II) and are central to the CANC2 vision system are reviewed.

1) **Descriptors**: Perceptual organization generates features.
2) **Recursive Grouping**: The grouping process is recursive.
3) **Invariance**: The geometrical relationships used for grouping are relatively invariant to changes in viewpoint.

A. Descriptors

Marr [46] views perceptual organization as grouping processes that operate on the "raw primal sketch" (primitive descriptors of the image) to build up descriptive primities. Witkin and Tenenbaum [72] state that the various structural entities identified by perceptual organization are "descriptions in terms of primitive features which brige the gap between the raw image and high-level knowledge and goals." We will use the term collated features for the groupings of image features identified by perceptual organization.

Collated features encode structural information. Vision informs us about the shape and spatial arrangement of objects in our environment, and thus, structure plays a crucial role in

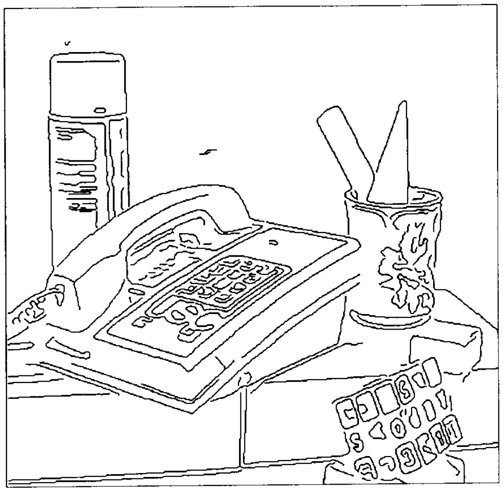


Fig. 3. Edges detected in Image I.

vision. Some of the descriptors obtained from a scene should describe the structural properties and relationships in it.

B. Recursive Grouping

Primitive image elements are grouped on the basis of their geometrical relationships. These representations or collated features can themselves be considered to be image tokens and collations of them formed based on the structural relationships among them. Thus, perceptual organization is recursive. It forms groupings on collated features obtained from previous groupings. Perceptual organization builds a hierarchy of collated features that capture different structural relationships and at different scales. These collated features are robust: When image elements are grouped on a particular relationship, the geometrical properties of the relationship provide information to overcome local problems in the detection of the primitive features [51].

The perceptual organization process is symbolic in nature. Various psychological experiments suggest that perceptual organization groups visual tokens on symbolic comparisons of their properties rather than direct visual measurements of the image [46], [65].

Collated features are formed by recursively grouping edges. Fig. 3 displays the edges detected in Image I (Fig. 2). Note that even in these fragmented edge maps, one can readily discern the objects and their structure.[2] Edges will be considered as purely geometric entities. Although contrast properties do influence the grouping process, they are not the essential factor because grouping can be performed in the absence of contrast, and they are not the deciding factor as grouping on contrast similarity is not preferred over grouping along structural relationships.

[2] Techniques for line-drawing interpretation [45], [69] are not applicable to these edge maps. Edges (or the edge contours obtained from contour tracing) and boundaries of regions cannot usually be used directly for interpretation as line drawings. Thus, current techniques for line drawing interpretation are not useful for obtaining structural information directly from edges or regions.

C. Invariance

An intelligent observer, in interpreting the visual stimuli provided by its surroundings, has to make consistent interpretations of a changing visual environment. To ensure consistency of the interpretations, one key task of the visual system would be to extract visual invariants, features, or descriptors of the visual data that do not change with changes in the parameters of observation, such as illumination and viewpoint.

The principle of nonaccidentalness [72], [43] states that regular geometric relationships are so unlikely to arise by accident that when detected, they almost certainly reflect some underlying causal relationship. Nonaccidental geometrical relationships are those that remain invariant over projections from different viewpoints. For example, parallel lines in 3-D always project (orthographically) as parallel lines in 2-D (invariance to viewpoint transformation). Invariance assigns causality. The probability of two lines, which are not parallel in 3-D, projecting as parallel lines in 2-D (due to accidental alignment) is so small that one can claim with high confidence that parallel lines detected in 2-D actually projected from parallel lines in 3-D.

Descriptors should be invariant to changes in viewing parameters. Viewpoint is one parameter that changes. Contrast may change due to changes in lighting, the reflectance functions of the surfaces, and the juxtaposition of objects. The detection of collated features is made robust (invariant) to changes in contrast by designing a detection mechanism that uses just the geometrical relationships among edges and not their contrast. Since the detection of edges is effected by contrast and edge interaction, the detection process for collations is made robust to local problems in edge detection by use of global structural information.

Additionally, the detected geometrical relationships are derived from the structures of the objects of interest. From an ecological standpoint, the human visual system must have evolved to be sensitive to the common, important object shapes around us. From a practical standpoint, a computer vision system needs to be sensitive to the structure of the objects it is designed to detect.

These two criteria (invariance to viewpoint transformations and correspondence to object structure) gives inferential leverage. When structural relationships that meet these criteria are detected, one can infer that the structure detected in the 2-D image projected from a related structure in 3-D (i.e., it is not due to accidental alignments) and that the 3-D structure corresponds to an object. Thus, such geometric groupings in the image have a high likelihood of corresponding to objects in the scene.

IV. Application

Collated features encode some of the geometrical or structural information available in a scene. They can be used as mid- and high-level abstract features by various visual modules:

- **Segmentation**: Structural information from various parts of the image is used for detecting collated features. This results in more robust detection in terms of overcoming local problems [48], [51]. As the collated features are relatively invariant to transformations in viewpoint and identify structural relationships among image features that also exist among object features, the collations have a high likelihood that they will correspond to objects.

- **Description**: Object shapes are described in terms of component shapes. This decomposition of shape description follows a structural basis where individual shape components are well formed and have simpler individual descriptions than that of the combined shape. The collations of features identify such individual structures that are useful in describing overall shapes. Collated features provide small local geometrical descriptions that can be combined to give global object descriptions. Therefore, when used for segmentation, collated features automatically provide shape descriptions of the segmented entities.
- **Attention Cues**: By the identification of significant features, collated features can act as attention mechanisms for visual inspection, guiding the detection of features at interesting locations in greater detail.
- **Correspondence**: Collated features are relatively invariant to viewpoint transformations. Thus, collated features are excellent tokens for correspondence when images of objects as viewed from different viewpoints have to be matched. In correspondence processes such as stereo, motion, and model matching, improved performances have been obtained by using more abstract features [47], [42], [50], [22], [43], [31]. Collated features have more attributes than edges, and thus, there is much less ambiguity in matching collated features than there is in matching primitive features. Their small number results in a significant reduction in the computational expense of matching.

V. CANC2

In the previous sections, the basic principles involved in the incorporation of perceptual organization into computer vision systems were discussed. The remaining sections present CANC2, which is a computer vision system designed along these principles. CANC2 generates a hierarchy of collated features for the generic visual domain of curved objects seen from any viewpoint. Fig. 2 shows an image of a scene from this visual domain. The collated features obtained by grouping edges on the geometric relationships of continuity, proximity, symmetry, and closure are curves, points, contours, symmetries, and ribbons. This feature hierarchy provides descriptors from the level of edges up to the level of 2-D shape descriptors of the surfaces that are visible in the image (refer to Fig. 4).

A. Geometrical Relationships

The Gestalt psychologists categorized the geometrical relationships humans perceptually organize on [43], [70] as the following:

- **Similarity**: Similar elements are grouped together.
- **Proximity**: Elements that are close together tend to be grouped together.

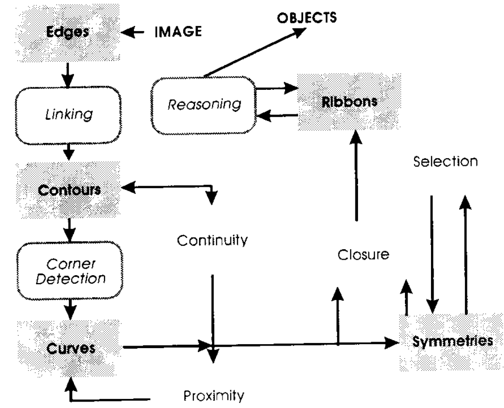


Fig. 4. CANC2 vision system.

- **Continuation**: Elements that lie along a common line or smooth curve are grouped together.
- **Symmetry**: Symmetric curves are grouped together.
- **Closure**: Curves are connected to enclose regions.
- **Familiarity**: Elements are grouped into familiar structures.

These relationships are not exclusive; groupings may be formed on some combinations of these relationships, and there may even be opposing groupings proposed by different relationships [39].

Precise representations of the tokens are required on which geometrical relationships can be computed. For simple tokens, such as points and lines, the tokens can themselves serve as the representations on which the relationships can be computed. For more complex tokens, the appropriate representations are not so obvious. Points and curves are the primary representations, i.e., complex tokens and collations, when represented as points and curves, can be grouped on proximity, continuity, symmetry, and closure to account for most of the (macro-structural) perceptual organization.

B. Collated Feature Hierarchy

The collated features generated by application of geometric organization on proximity, continuity, symmetry and closure on intensity edges are as follows:

- **Curves**: A curve is a smooth curve through contiguous edges (with possibly some gaps) with no corners (tangent discontinuities or extrema of curvature). Curves are obtained by grouping edges on continuity and by grouping curves on proximity.
- **Points**: Points are terminations and junctions of curves. Points are detected by considering local junctions between curves.
- **Contours**: A contour is an ordered set of contiguous curves.
- **Symmetries**: Symmetries are pairs of mutually symmetric curves.
- **Ribbons**: A ribbon is the area enclosed by two symmetric curves. It is described by a symmetry axis and a sweeping

rule. The sweeping rule gives, for each point on the axis, the corresponding pair of symmetric points on the two curves. Ribbons are detected by searching for closure at the ends of symmetries.

Cocurvilinearity and symmetry are relatively viewpoint invariant. Coterminations (junctions) and cocurvilinearity among 3-D curves project with the same relationship in 2-D [4], [43]. Although curvature extrema are also considered to be junctions or corners, they may change specially in foreshortened views. However, people have been shown to be sensitive to curvature extrema [19].

The features in this representational framework are also motivated by their utility to various facets of computer vision. Surface boundaries of most objects are composed of smooth curves. Many natural shapes and man-made objects exhibit symmetry. Curves and the related features (points and contours) are more abstract features than edges and correspond to physical entities such as object and surface boundaries (as well as surface markings, shadows, etc.). Ribbons are a popular primitive for describing 2-D shapes [10], [11], [55], [61]. The symmetry axes are useful for describing ribbons and for making shape interpretations.

This hierarchy of collated features is suitable for handling objects whose projected surfaces can be described by combinations of ribbons. Because ribbons are commonly used shape representations for generic 2-D shapes, this is not overtly restrictive. However, it may exclude amorphous objects such as clouds and foliage.

VI. CURVES

A curve is a smooth contour that has no corners. All contours are segmented into component curves at corners.

A. Cocurvilinearity

Cocurvilinearity is the structural relationship on which image tokens are grouped into smooth curves. Cocurvilinear tokens in a scene are mapped to cocurvilinear tokens in the 2-D image [43]; thus, this relationship is invariant to viewpoint transformations.

Most of the previous work on cocurvilinearity has concentrated on detecting geometric organization among dot patterns [65], [32], [67]. Cocurvilinearity of curves can be broken into two components: continuity and proximity of similarly oriented curves. Fig. 5 shows cocurvilinearity grouping primarily on continuity.

Curve grouping alone, without recourse to perceptual organization at the scale of object structure, can never give curves that are correct in a global context. The curve groupings computed by the system encode structural information at the curve level but not the object level, and the final curves accepted are those belonging to selected ribbons at the end of the complete grouping process.

B. Continuity

Boundaries are smooth and get projected as smooth curves in the image (termed "continuity of discontinuities"[46]).

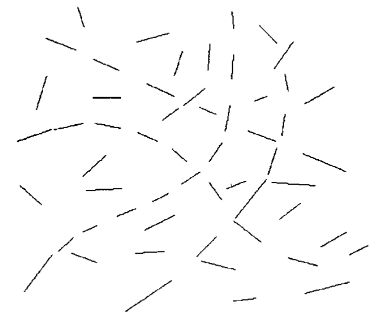


Fig. 5. Curve segments.

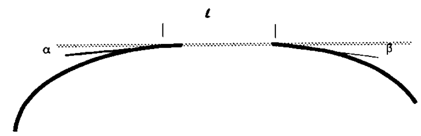


Fig. 6. Calculation of bending.

Collations of curve sections, in the form of smooth curves, are detected by a local, noniterative process that selects the most collinear, or least bent, joins among neighboring tokens. This formulation of continuity-based organization is influenced by the experiments by Stevens and Brookes [65] and Zucker [74], which indicate that the grouping process can be modeled by local selection of most collinear pairings.

For each curve end c_1, joins to all the curve ends that lie in its neighborhood (10% of the image size) are considered. These may be further restricted to a wedge $\pm\theta^\circ$ about the tangent at that curve-end c_1. A measure of bending $E_B = (\alpha^2+\beta^2)(a+bl)$, a, b contants (see Fig. 6), which also tries to account for proximity by favoring short joins, is calculated for each join considered. The parameter a controls the departure from collinearity of the joined curves, and b controls the sensitivity to the lengths of the gaps being filled in. For all the images processed, $a = 1.0$ and $b = 0.1$ were used. The join with the minimum bending energy E_B and the joins with bending within some factor of it (three times the minimum) are selected. Next, only those joins that are selected by both the curve ends that the join connects are retained. These remaining joins may be further thresholded on a maximum acceptable bending level. Fig. 7 shows all the joins that are formed on the curves in Fig. 5 within 30° about each curve end. Fig 8 shows the joins that remain after selection of compatible joins within a factor of three of the minimum.

1) Implementation: Fig. 3 displays the edges detected in Fig. 2 using a Canny edge detector [12]. Edges are linked into contours on the basis of eight-neighbor connectivity. These edge contours contain wrong linking at forks (which cannot be seen by a visual inspection of Fig. 3) linking errors caused by edge displacement at junctions and edge dropouts due to poor contrast.

The edge contours so obtained are segmented into curves at curvature extremas. Adaptive smoothing [63] is employed to detect curvature extremas (Fig. 9). Chains of curves joined

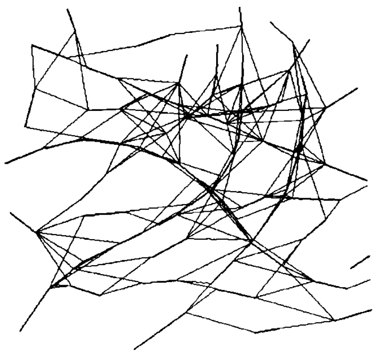


Fig. 7. All the joins considered.

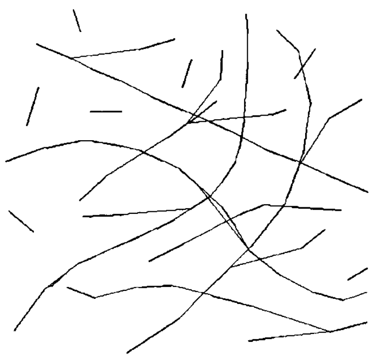


Fig. 8. Selected least bend joins.

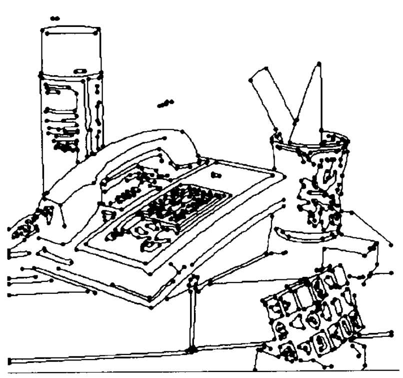


Fig. 9. Corners detected in the edge contours of Fig. 3.

by the continuity process are rerepresented as edge contours. In these contours (see Fig. 8), most of the problems of small gaps or wrong linking, which is present in the original edge contours, have been rectified. The grouped curves from one level of recursion are themselves considered to be individual curves for the next pass, thus bridging larger gaps and/or sharper bends than were acceptable in earlier passes (i.e., bringing more global structural information to bear on the grouping process). The original curves that go into a grouped curve are also maintained as alternate descriptors.

A continuity-based organization process, which is similar to that defined on curves, is also defined among point tokens

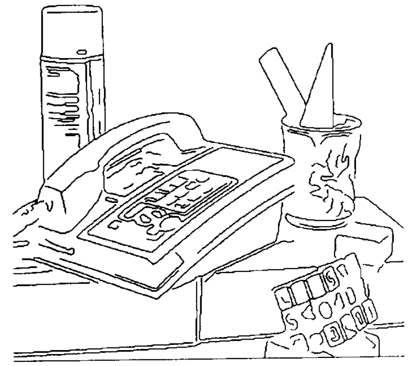


Fig. 10. Joined curves—cocurvilinearity.

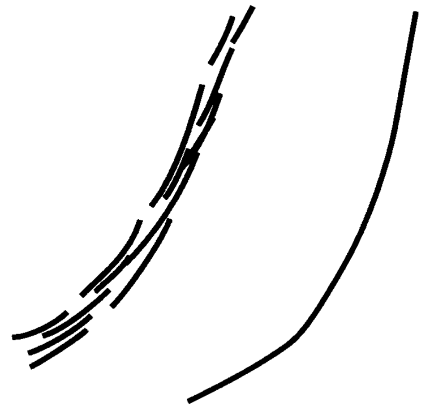


Fig. 11. Cocurvilinearity on proximity.

[49]. Point groupings have not been used in this paper.

C. Proximity

Overlapping, proximate, similarly oriented curves are grouped into curvilinear structures. Consider Fig. 11. The bunched curves in the left part of the figure, which can be seen as separate curves, can also be seen as a single curvilinear structure when examined at the scale of the curve on the right. This grouping is motivated by issues of scale (Fig. 12); proximate parallel curves are interpreted as boundaries of surfaces (or ribbons) rather than as representing multiple, narrow surfaces.

1) Implementation: For each curve overlapping, nearly parallel curves are detected in its neighborhood. Curves lying within a distance n are considered. Two curves are parallel if the orthogonal distance from the symmetry axis (the complete axis is not computed; it is sufficient to have a few points on the axis) to each curve is equal $\pm\epsilon$ at different positions along the axis. For all the images processed, $n = 15$ and $\epsilon = 0.3n$ were used. The value of n used depends on the scale, which is the contents of the scene. Estimation of the value of n, in general, would require an analysis of the distribution of distances between parallel curves in the image.

Next, equivalence sets of curves are formed on the rela-

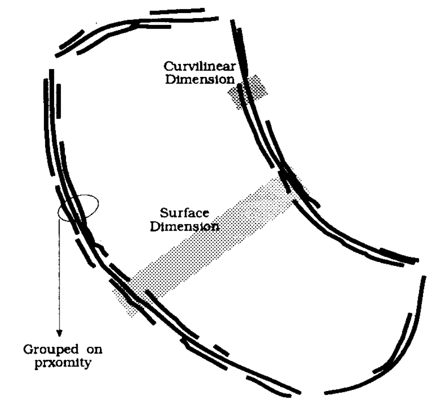


Fig. 12. Scale in proximity-based grouping.

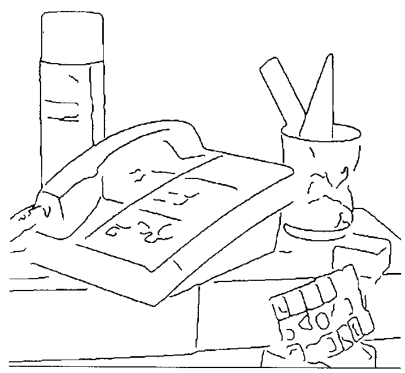


Fig. 13. Curves obtained after grouping on proximity.

tionship of proximate parallelism. Each set is a grouping of co-curvilinear curves and is simply represented by the longest curve in it. Proximate co-curvilinear groupings found in the curves of Fig. 10, and so represented, are shown in Fig. 13.

The structure of such a grouping is a single curve at a gross scale but breaks up into smaller curves when examined in detail. Thus, while Fig. 13 is a simpler representation of the curves in Fig. 10, some structural detail may be lost (for example, the thickness of the handle of the handset). This collated features is useful in providing simple, coarse descriptions of a scene. If details are needed at some stage, the original curves that formed a curve (grouping) are available for examination.

D. Co-termination

Co-terminations are junctions between curves. Curves are extended by straight lines tangent to the curve-end and extending out n pixels, where n pixels is the width used for proximity grouping for that image. Only long curves ($> n$ pixels) are considered for extension. The nearest cotermination for a curve end is located along the extension at that end. If a junction is so detected, the incident curves are extended up to that junction. Fig. 14 shows the curves form Fig. 13 after extension, and the resulting coterminations (and terminations).

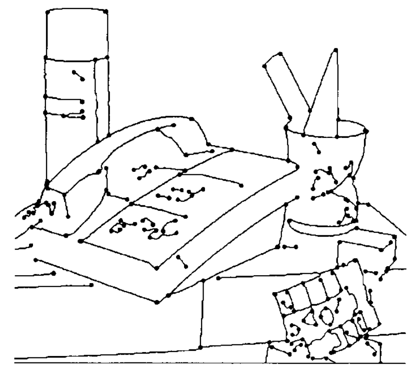


Fig. 14. Extended curves and coterminations.

VII. Symmetry

Symmetry: Symmetry is defined as a one-to-one mapping (if the curves are considered infinitesimally divisible) between the points of two curves. The symmetry axis is defined as the locus of the midpoint of the straight lines (or, for generality, some curve) joining a point on one curve to its image in the other.

Symmetry is a characterization (measure) of the structural relationship between two curves. The structural relationship between two curves that bound a surface or an object is likely to be more regular and organized than the relationship between structurally unrelated curves, for example, relationships between the boundaries of different objects or relationships between texture curves and bounding curves.

Although the bounding curves of most surfaces exhibit some structural regularity, the symmetry relationship between them may not be of a specific form such as parallel or bilateral.[3] Even when such specific symmetries are present, they may be difficult to confirm due to the distortions in the imaging process. In this paper, a more general symmetry relationship is defined, and a quantitative measure is assigned to the regularity of the relationship based on the geometry of the symmetry. This quantitative measure of the symmetry is indicative of the likelihood of two symmetric curves to be physically related (for example, as boundaries of the same surface), in comparison with the other possible symmetry pairings of these curves.

A. Previous Work

The types of axes defined, and the literature on their properties and their detection, is vast; we will not attempt to present a complete account of them. A good survey and analysis of symmetry axes, especially in the context of ribbons, can be found in [61]. Rao [59] presents a survey of symmetry axes in the context of generalized cylinders.

There are three important issues (detection, description, and application) that have to be considered in any discussion of symmetry axes:

[3] *Parallel Symmetry*: The tangents directions at the symmetric points on the two curves is the same. *Bilateral Symmetry*: One curve is the reflection of the other across a straight axis.

- **Description**: Most proposals of symmetry axes have been for the axial description of 2-D areas [5], [10], [8], [9] and 3-D volumes [4], [55], [54], [52]. Axial shape representation is useful for providing a simple shape description of an object, for providing an object-centered description frame, and for the recovery of the shape being described by the axis.

Blum [5] defined the symmetric axis transform (SAT) as the locus of the centers of maximal disks inside a given 2-D shape. It suffers from sensitivity to small changes in boundary, and sometimes parts of the axes may lie outside the object. Brady [8], [9] modified the symmetry axis transform to smoothed local symmetries (SLS) by placing the axis at the mid point of the chord joining the corresponding points (rather than the center of the circle), thus restricting the axes to be local or near to the object parts they defined. Nackman and Pizer [52] extended Blum's SAT to 3-D volumes, but the resulting axes can be surfaces.

Binford [4] defined generalized cylinders (GC) as the volume defined by sweeping an area along an axis. The GC axes for a volume are not unique.

Shapes from Blum's and Brady's axes are usually recoverable, whereas those from Binford's axes may not be recoverable [61]. For Brady's axes, and usually for implementations for generalized cylinders [53], [59], [10], the symmetry axis is orthogonal to the sweeping rule.

- **Application**: Although shape description itself is one application for symmetry axes, there are other applications such as model matching [53], [11] and 3-D shape interpretation (shape from contour) based on the concept of skew symmetry [33], [34], [68].
- **Detection**: In the computation of SAT [5], SLS [9], [13], and generalized cylinders [53], it is assumed that closed boundaries are available. The pairing of symmetric contours is not an issue as no texture markings and other objects in the scene are considered (Nevatia [53] handles the case of multiple objects by considering only the nearest symmetries). In the systems for shape from contour, the symmetry pairings of the curves is assumed to be known. Brooks [10] uses prediction from the models and viewpoint to help in the detection of ribbons. Rao [59] has extended Nevatia's work for images with markings and broken contours.

There are important deficiencies in a broad class of these axes:

Axes defined in the context of generalized cones [4], [1], [55] are suitable for symmetry relationships between limb or extremal boundaries of (straight homogeneous) generalized cones and not for boundaries arising from orientation discontinuities.

The line joining symmetric points is restricted to being orthogonal to the symmetry axis [53], [55], [8], [13], [10], [59], and thus, cases of skew symmetry are not handled.

In the projected scene, the axes may change with change in viewpoint and may not correspond to the projected axes of the object. In addition, the axes for a given shape may be disjoint.

Detection of these axes is based on matching points on the contours, i.e., for each point on a contour, possible matches to all the points on the other contour have to be considered. Thus, these detection techniques have a computational complexity of $O(n^2)$, where n is the number of edges on a contour (this can be reduced to $O(nk)$, where k is constant, by quantizing the search space, as proposed by Nevatia [53]). Point-by-point matching means utilizing some measure, such as tangent direction, at each point to evaluate the match. Not only are such measures noisy along real edge contours, but they usually do not vary much along a curve, making the localization of matches difficult. A better scheme would require matches at well-localized positions, such as corners, along the contours. The situation is analogous to stereo matching, where it is found that matching locations of sharp change, i.e., intensity discontinuities or zero crossings, is better than matching intensity pixels.

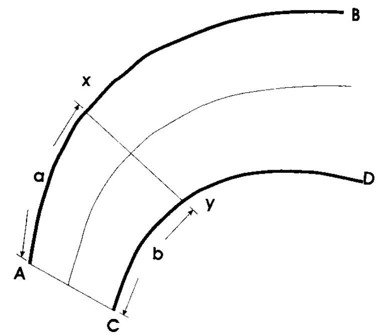


Fig. 15. Transformationally invariant symmetry axis.

B. Transformationally Invariant Symmetry Axes

The symmetry axis between two curves is defined as the locus of the midpoints of the lines joining points at equal length ratios along the curves. Consider Fig. 15. Let the length of curve AB be s_1 and that of CD be s_2. A point x on AB is mapped to a point y on CD if and only if

$$\frac{a}{s_1} = \frac{b}{s_2}$$

where a is the length of curve section Ax, and b is the length of curve section Cy.

For the detection of this axis between two contours, only the corners of the contours need to be matched because the match for the points between the corners is automatically defined. Thus, detection of this symmetry axis involves matching of curves rather than edges.

The axis is also invariant to viewpoint transformations for important classes of curves and specific symmetry relationships. Consider imaging situations with linear transformations such as orthographic projections and limited perspective:

$$[T]\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \alpha_1 x + \beta_1 y + \gamma_1 z \\ \alpha_2 x + \beta_2 y + \gamma_2 z \end{bmatrix}$$

Midpoints map to midpoints in the projection:

$$\begin{bmatrix} \frac{1}{2}((\alpha_1 x_1 + \beta_1 y_1 + \gamma_1 z_1) + (\alpha_1 x_2 + \beta_1 y_2 + \gamma_1 z_2)) \\ \frac{1}{2}((\alpha_2 x_1 + \beta_2 y_1 + \gamma_2 z_1) + (\alpha_2 x_2 + \beta_2 y_2 + \gamma_2 z_2)) \end{bmatrix} =$$

$$\begin{bmatrix} \frac{1}{2}(\alpha_1(x_1+x_2)) + \frac{1}{2}(\beta_1(y_1+y_2)) + \frac{1}{2}(\gamma_1(z_1+z_2)) \\ \frac{1}{2}(\alpha_2(x_1+x_2)) + \frac{1}{2}(\beta_2(y_1+y_2)) + \frac{1}{2}(\gamma_2(z_1+z_2)) \end{bmatrix}.$$

Therefore, the midpoints of the lines joining symmetric points are invariant under linear transformations. If, in addition, the length ratios of curves are maintained under the transformation, then the axis would be invariant to these transformations. In general, the length ratios of curve lengths are not maintained under all linear transformations. However, for the special case of straight lines, the length ratios are maintained under any linear transformation (for a proof, replace $1/2$ by any fraction in the above equations). Thus, for straight lines, this symmetry axes is invariant to viewpoint transformations.

The axis is also invariant to viewpoint transformations for curves for some restricted but important types of symmetries. For example, Ulupinar and Nevatia [68] have proposed two specific symmetries that are useful for determining shapes of surfaces (shape from contour). In case two curves have either of these symmetry relationships, then the symmetry axis defined here corresponds to it for all viewpoints.

C. Implementation

For each curve, possible symmetry relationships are considered with every other curve in the image. Some heuristics are used to limit the number of symmetries:

1) The shorter of the two curves should be no less than one third the length of the longer curve.
2) There should be some overlap between the two curves. Using only the correspondence between the curve ends, a straight symmetry axis is proposed, and at least one third of either curve should project, orthographically, onto this axis. The minimum amount of overlap required limits the maximum amount of skew we are ready to entertain.

Curve pairs that meet these two requirements are hypothesized as possibly being symmetric. However, the symmetry axes between them is not calculated (as measurements for evaluating the symmetries can be made by just knowing a few positions along each axis), and the axes are provisionally represented by their simpler straight line counterparts. Fig. 16 shows all the symmetry axes hypothesized for the curves in Fig. 14.

1) Discussion on Symmetry Axes: The cocurvilinear grouping of curves aids symmetry detection. Reciprocally, cocurvilinearity of axes of symmetry can guide cocurvilinear grouping of their constituent curves (this has not been implemented).

The proposed symmetry axis is sensitive to the location of corners and junctions. Corners located at curvature extremas are largely invariant to viewpoint transformations but may change on extreme foreshortening. T junctions are dependent on viewpoint and thus adversely effect symmetry axes. However, symmetry probably does not play a role in the completion of occluded boundaries. Inspect Fig. 17. Although the object appears (bilateral) symmetric and the symmetry is emphasized by marking the symmetry axis, the occluded portion of the contour is completed using continuity and not symmetry. The detection of symmetry axes could be made robust to occlusion by using other techniques to account for the occluded curves.

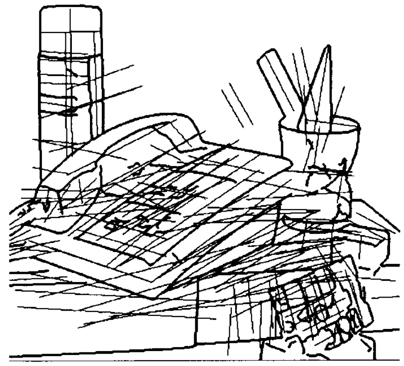


Fig. 16. All the symmetry axes hypothesized.

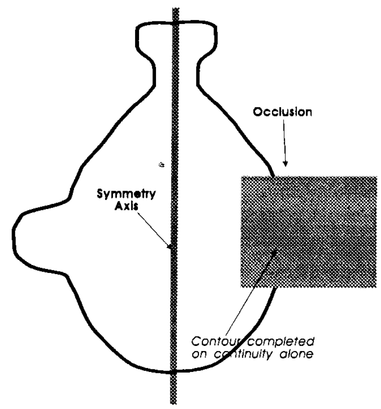


Fig. 17. Symmetry is not used to overcome occlusion.

The current system does not perform special processing to handle occlusion.

VIII. Evaluation and Selection of Symmetries

The detection of collated features, where all reasonable groupings among tokens results in the formation of collated features, is followed by selection, where only the more suitable collated features are retained. The detection and selection processes could proceed simultaneously.

At each level of the collated features hierarchy, various collations are in contention as they provide alternative groupings of the underlying tokens. In addition, some collations may have been formed on weak evidence (evidence that seems too weak when compared with that for other collated features at that level). A selection process has to choose good collations, i.e., those that have a high likelihood of corresponding to individual object parts.

The goodness of a collated feature depends on how it compares with its alternatives in terms of the support or contradiction from its component primitive features and other related image features.

The problem of selecting the best set of collations can be formulated as selecting the best set of hypotheses, given relationships of support and conflict among them [48]. Consider the following:

- A set of **Hypotheses** $H = h_i$.
- A unary function **Value** $V(h_i) = V_i, 0 \leq V_i \leq 1$, which assigns a confidence level to each hypothesis.
- A binary function **Support** $S(h_i, h_j) = T_{ij} V_i V_j, T_{ij} > 0$, and its value depends on the support relationship between two hypotheses.
- A binary function **Conflict** $C(h_i, h_j) = T_{ij} V_i V_j, T_{ij} < 0$.
- A unary function **Input** $I(h_i) = I_i$, which is a sum of the evidence or measurements for the hypothesis h_i .
- A function **Input Evidence** $E(h_i) = I_i V_i$, which measures the contribution to the the total evidence from hypothesis h_i.

To choose the best consistent set of hypothessis that maximize evidence, we wish to assign confidence values to hypotheses such that $\sum S + \sum C + \sum E$ (note the terms in C are negative) is maximized. Rewriting the above term, we have $\sum \sum T_{ij} V_i V_j + \sum I_i V_i$. Our goal is to find the optimal feature groupings consistent with the known optical and geometrical constraints [3], [15]. Note that all the constraints must be simultaneously satisfied to reach global consistency across all levels of the hierarchy.

One parallel technique to solve this problem is relaxation where a cost function associated with the network is minimized. We wish to select the best consistent feature groupings and reject the bad groupings. If we formulate the cost function such that the optimal solution corresponds to its global minima, then the problem of locating the best groupings reduces to that of optimizing the cost of the network given the constraints (defined by the relations) between the collated features and the observed image characteristics. Parallel optimization techniques such as simulated annealing [38], Hopfield networks [28], [29], Boltzman machines [15], [62], [16], probabilistic solutions [23], and connectionist methods [62], [18], have been proposed for such problems.

A. Constraint Satisfaction Networks

The collated features and the relationships of support and conflict among them naturally define a network with the collations serving as nodes and the relationships as arcs. We use the Hopfield network to implement this network as a constraint satisfaction network. Following the notation convention of Hopfield and Tank [27], [26], we describe the behavior of each node in the network by

$$\frac{du_i}{dt} = -u_i + \sum_{j=1}^{N} T_{ij} V_i + I_i - h_i \qquad (1)$$

$$V_i = g(u_i) = \frac{1}{2}(1 + \tanh(u_i)).$$

In the above eauations, N is the total number of nodes, T_{ij} is the weight on the link from node j to node i, I_i is the total input to node i, V_i is the output of node i, and u_i is the membrane potential of node i. The gain function g is sigmoidal. The addition of h_i, which is the resting potential or bias, is useful in adjusting the sensitivity of a neuron by shifting its gain curve. For purposes of analysis of the network, the resting potential may be combined with the input.

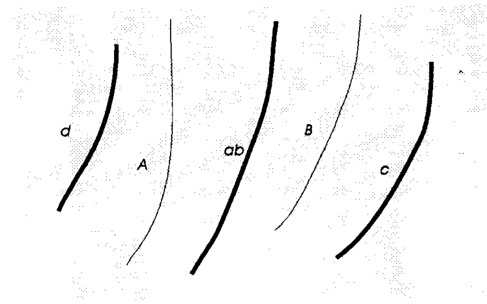


Fig. 18. Competition between symmetries.

When the network has symmetric connections, i.e., $T_{ij} = T_{ji}$, the network, where each element has the above equation of motion, converges to stable states. This property has also been shown for more general networks by Hummel and Zucker [30]. When the gain function g is high gain (width of the gain curve is narrow), the stable states of the N elements are the local minima of the following cost function with the outputs of the nodes at 0 or 1 [26]:

$$E = -\frac{1}{2} \sum_i \sum_j T_{ij} V_i V_j - \sum_i V_i I_i. \qquad (2)$$

Note that minimizing the above term is equivalent to maximizing the term given the previous subsection (after replacing T_{ij} by half its value in that term). The signs in the above cost function suggest that if we wish to select mutually supporting collations and reject mutually conflicting collated features, the weights T_{ij} between supporting hypotheses should be positive and that between conflicting hypotheses should be negative. Those optical and geometrical constraints, which are not expressed purely via the interrelationships between the interpretations, should be fed as inputs I_i to the nodes. Again, the sign in (1) and (2) shows that supporting evidence should be included as positive input and contradicting evidence as negative input.

In Fig. 18, consider the symmetry relationship between the curves A and B represented by the symmetry axes ab. This symmetry axis lies to the right of A and to left of B and is thus competing with all the symmetry axes resulting from a symmetry relation with the right side of A, such as axis c, and the axes formed by symmetry relations with the left side of B, such as axis d. Symmetry axes for alternate representations of a curve, which lie on the same side, are also competing.

Each side of a curve can bound, at most, one surface. Ideally, at most, one symmetry axis should be selected for each side of a curve. To avoid missing some weak but correct symmetries, it is desirable to allow more than one symmetry to be selected. The curves resulting from texture or surface markings are usually nearer to a boundary curve than the corresponding boundary curve, and thus, this relationship benefits from a better aspect ratio. In the case that a boundary curve is fragmented, the short amount of overlap and the difference in lengths between the symmetric curves would put

TABLE I
EFFECT OF CHANGE OF WEIGHTS ON SELECTION

Change in weights	Number of symmetries selected	number of correct symmetries missed
0%	55	none
+100%	55	none
+200%	59	none
-50%	54	none
-75%	54	none

a low evaluation on the symmetry relationship (axis). For these reasons, it is better to be not too critical in the selection of the best axis, i.e., force the selection of one unique axis for each curve side, but to allow for the selection of a few best axes.

In the network, each axis is represent by a node (or a neuron). Each of its competitors is connected to it by a negatively weighted link. The evaluation of the axis (described below) is fed as input to the node for that axis. The network converges after a few iterations (about 5). Each iteration represents one time constant and is itself implemented as 10 subiterations. The nodes with high output (> 0.8) are considered to be selected, whereas the rest are rejected. Note that in this network, the nodes or "neurons" represent high-level features [7].

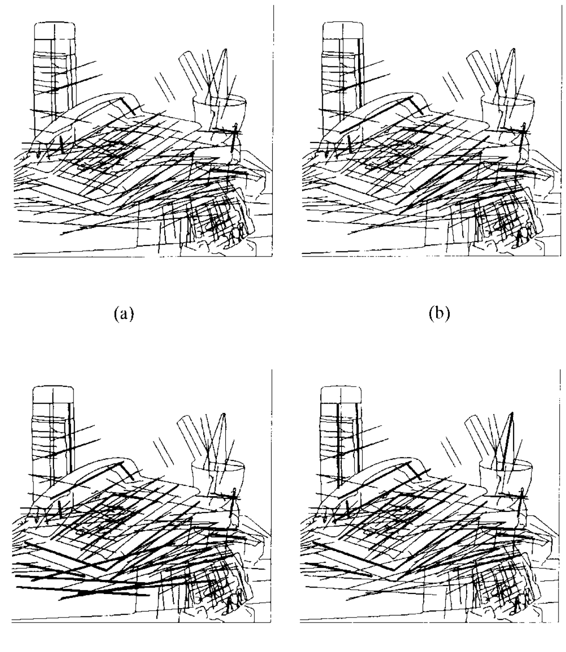
The value of each axis is computed as a weighted sum of a numerical representation of the following measures:

1) For an axis, the amount of cover described as the length of the contours for which it is the axis
2) the aspect ratio defined as the ratio of the axis length to the distance between the two curves
3) the similarity between the length of the two symmetric curves measured as the difference between the length of the two curves, normalized by the length of the longer curve
4) amount of skew between the curves
5) parallelism between the two curves
6) parallelism between the ends of the curves
7) length of the axis.

The value for an axis is computed as ($8.0 \times$ *aspect* $- 15.0\times$ (angular difference between the two curves in radians) $-4.0\times$ (normalized length difference) $-3.0\times$ (angular difference between the ends) $-5.0 \times$ *skew-angle* $+ 2.0 \times$ *length*) and is fed to its node as input. The numbers used in the linear combination of the measures are chosen to normalize the different units used for the different geometrical measures and to assign relative weighting to the different measures in accordance with their subjective importance. The same weights were used for all the images processed. Table I shows the robustness to change in weights (for Image I).

The constraint satisfaction network is used to prune the number of collations. Structural evidence, from the monocular reasoning phase of the segmentation process, finally decides whether a given pair of symmetric curves does bound a surface.

Figs. 19(a) through 20(d) show snapshots of the axes-selection process by the constraint satisfaction network. The axes that correspond to nodes with high output at subiteration are shown by thick lines. The symmetry axes that get selected



(a) (b) (c) (d)

Fig. 19. Selection of symmetry axes by CSN: (a) After 2 iterations; (b) after 3 iterations; (c) after 4 iterations; (d) after 5 iterations.

earliest are those for which there are no competing alternates (Fig. 19(a)). Next, those axes with high evaluation, especially those with high aspect ratios, get selected (Fig. 19(b)). Some axes, which get selected during the early stages of the relaxation process (Fig. 19(c)), get suppressed later (Fig. 19(d)). Although some of these axes are subsequently selected (Figs. 20(a) and (b)), some of them may not get selected at all (Fig. 20(d)). The network stabilizes after the 25th subiteration (Fig. 20(d)), and there are no new axes selected (or rejected) in subsequent iterations.

In addition to the symmetries selected by the constraint satisfaction network, those symmetries where the symmetric curves are joined, at least at one end, by a single curve are also selected. This structural relationship ensures that symmetry axes that have low input (due to low aspect ratios or high skew) but high chances of corresponding to surfaces (due to simple closure at one end) also get considered for forming ribbons. Fig. 21 shows these axes, which we call u axes. Note that most of the u axes also get selected by the constraint satisfaction network.

Fig. 22 shows the axes selected from Fig. 16 (union of the axes selected by the CSN and those marked as u axes). The selected axes have been completely computed.

IX. RIBBONS

Closure: There is a tendency for curves to be completed so that they form enclosed regions. Each selected symmetry axis describes a ribbon or the area enclosed by the pair of symmetric curves (see Fig. 23). More than one symmet-

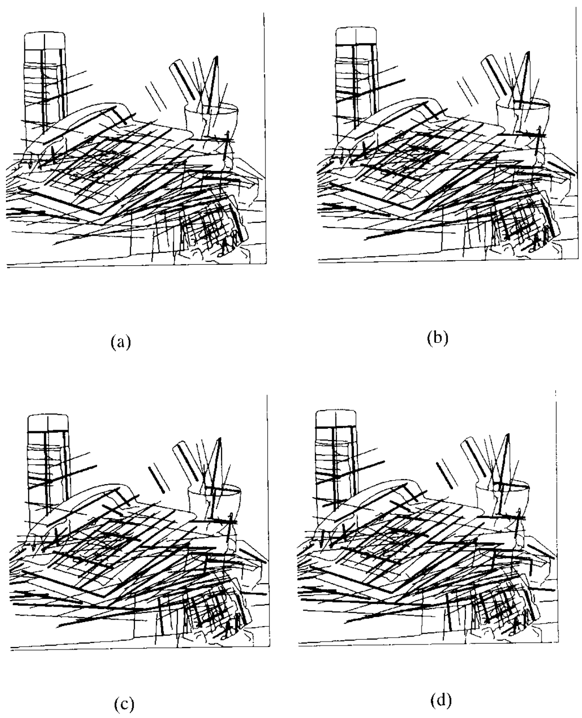


Fig. 20. Selection of symmetry axes by CSN: (a) After 6 iterations; (b) after 8 iterations; (c) after 13 iterations; (d) after 25 iterations.

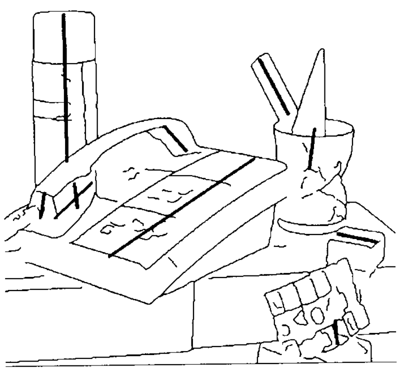


Fig. 21. U axes.

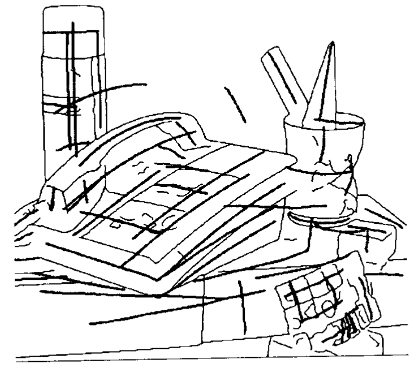


Fig. 22. Selected axes.

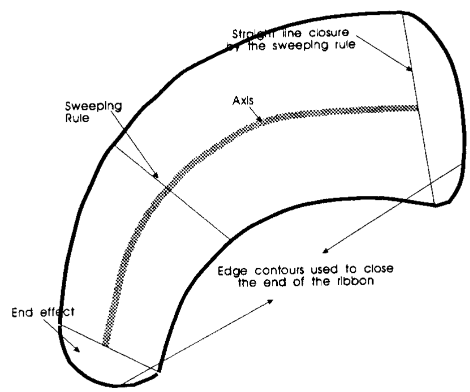


Fig. 23. Ribbon.

ric curve may have been selected for a curve, resulting in overlapping ribbons. This can be viewed as a hierarchical description scheme, in which the area described by one ribbon at one level, may at a finer level, be described by multiple ribbons. Alternatively, the overlapping ribbons can be viewed as providing alternate descriptions of the same underlying structure.

Since a ribbon describes an area, it needs to be enclosed in a boundary. The symmetric curves provide two boundaries. The other boundaries are provided by straight lines joining the ends of the curves (i.e., the symmetric points at each end of the axis). In real images, there may be edge contours closing the ends of the ribbons. For practical applications, it would be more suitable to use these edge contours, wherever present, for closing the ends of the ribbons. However, the shape described by the axis and sweeping rule still describes only the area obtained by assuming the ends closed by straight lines, but it does not account for the end effects caused by the differently shaped edge contours. This may be rectified by using curves to describe the sweeping rule along the axis, but it would result in a much more complex shape description.

A. Description

Two structural properties of 2-D shapes to which our visual system gives special attention are symmetries and axes. The axes correspond to coordinate axes tied to the object and mark an important step in our visual process of moving from observer-centered descriptions to object-centered descriptions [46]. The axes often correspond to some axes of symmetry, but other factors such the gravitational direction or context from other shapes play a role [56]. The representation axes also have a related sweeping rule. Complex shapes are often segmented into simpler shapes when there is a change in the sweeping rule (they also correspond to corners in the bounding curves).

The decomposition of complex shapes into simpler shapes and the representation of these component shapes in terms of an axis of elongation and a sweeping rule serves as a description of the 2-D shape that mimics generalized cylinders

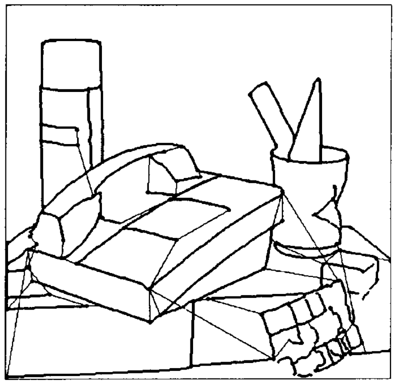


Fig. 24. Original closings for ribbon ends.

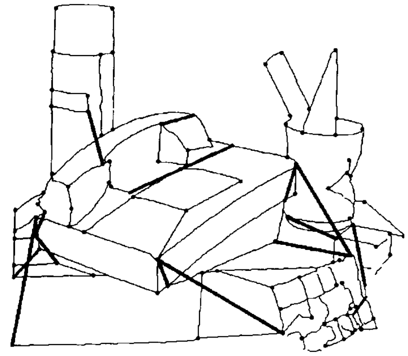


Fig. 25. Wrong joins.

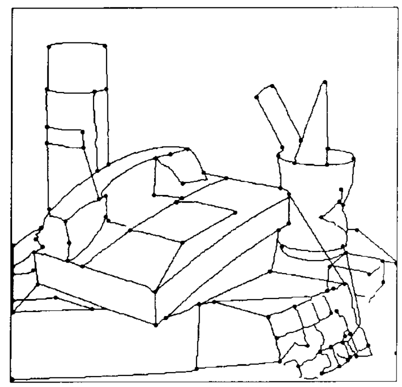


Fig. 26. End closures recomputed.

for 3-D shapes: a form of 2-D generalized cylinders, which is also called ribbons (Fig. 23). Thus, the ribbons that segment a scene into visible surfaces (Section X) also provide a shape description for those surfaces. The symmetry axes detected act as object-centered axes for the description.

B. Closure

The next step is to compute closure for the ends of the ribbons, with contours composed of the curves detected in the image. These enhanced ribbons (which we shall simply call ribbons from now on) are useful for segmenting scenes into visible surfaces. Some surfaces may have complex shapes and may be described by a combination of ribbons. For each surface, the component ribbons provide a shape description for the surface. The search for contours closing the two ends proceeds in the following order:

If the two symmetric curves form a corner, then that junction forms the closed end.

The curves are represented by a graph structure, with the curves represented as arcs and the points (curve terminations and junctions) represented as nodes of the graph. A best first search is done on this graph structure between the two nodes corresponding to the two terminations of the symmetric curves bounding a ribbon end for a path. The path is represented as the contour closing that ribbon end.

The curves lying in the areas between the two points are examined. If some curves are present, they are grouped on continuity to compute a contour closing the ribbon end.

If no contours are found (this may be due to either missing edges or complex shaped surfaces, where a single surface may be represented by more than one ribbon), a straight line join between the two curve ends is proposed as the closure for that ribbon end. Ribbons so formed are shown in Fig. 24.

X. Segmentation

The curves, points, symmetries, and ribbons detected form hierarchies of collated features describing objects in a scene. These collation hierarchies can be employed for segmentation of the scene into visible surfaces and objects. This segmentation is achieved by reasoning on the geometrical relationships between the various collated features, usually at the level of ribbons. Although the segmentation process itself does not employ perceptual organization, in most applications (for example, Section XII), collated features would first be used for segmentation to prune physically unlikely collations.

A. Surface Segmentation

1) Opacity: During ribbon formation for those ribbon ends where no candidate edge contours were found for closure, straight lines were proposed as closures. Assuming that the scene is composed of opaque surfaces, straight-line closures that cross over edge boundaries of other ribbons are in error (this is shown in Fig. 25 by bold lines). A search is conducted on the augmented curve graph for possible paths that could replace these erroneous closures. Fig. 26 shows the ribbons with the closures so recomputed. Any ribbons that still have ends that cross over other ribbons (wrong closure) are rejected.

2) Edge Support: Ribbons with substantial parts of the boundaries represented by hypothesised straight lines that have neither edge support nor form-shared boundaries with another viable ribbon are rejected. A ribbon has poor closure if either of the end closures is more than two thirds the length of the longer symmetric curve and has less than one tenth of its length supported by edges. The ribbons shown in Fig. 27 are those left after rejecting those ribbons that have wrong closures or poor closures.

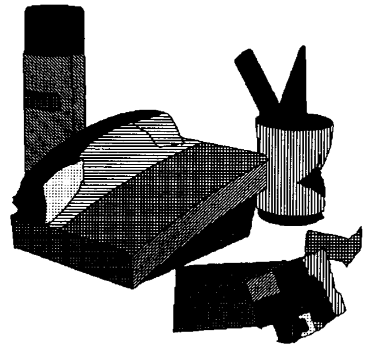


Fig. 27. Selected ribbons.

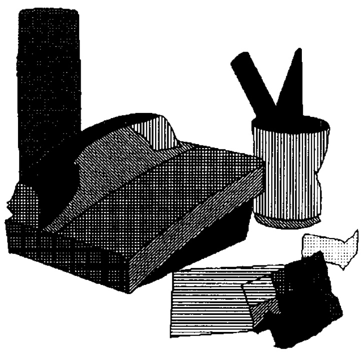


Fig. 28. Ribbons (after removing false T junctions).

3) Occlusion: Assuming general viewpoint and low probability of accidental alignments, a T junction indicates occlusion, with the stem of the T being occluded by a surface whose boundary includes the top of the T (the stem of T obviously does not lie in this occluding surface) [4]. Therefore, if ribbon A is enclosed inside a bigger ribbon B with at least one boundary of A not shared by B terminating at a T junction with B's boundary, then unless B is found to be occluded (along the boundary section contributing to the T junction), this configuration does not support the occlusion interpretation. Thus, ribbon A can be rejected as it forms a false T junction (the ribbon A most likely resulted from surface markings on the surface enclosed by ribbon B). Ribbons that remain after removing those that form false T junctions are displayed in Fig. 28.

B. Object Segmentation

The ribbons detected in an image correspond to the visible surfaces of the objects in the scene. Based on the structural relationships between the ribbons, the objects in the scene may be inferred. The premise is that as the objects are randomly placed in the scene, the relationships between the surfaces of any one object are more regular or organized than the relationships between the surfaces of different objects. Two ribbons (representing surfaces or parts of surfaces) belong to the same object if the following conditions are satisfied:

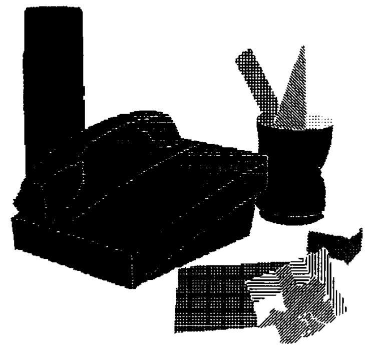


Fig. 29. Objects detected in Image I.

- The two ribbons share a complete side. A complete side of a ribbon is either one of the symmetric curves or one of the contours closing the gaps.
- One is a component of the other. A ribbon A is a component of ribbon B if it lies completely inside B and shares a part of the boundary of B.

These two conditions, when met, define the belong-to-the-same-object relationship. This relationship is an equivalence relationship, and the resulting equivalence classes correspond to the objects. The two conditions ensure that adjoining ribbons have a nonaccidental relationship before being considered as parts of the same object. These two conditions are not guaranteed to be correct for all objects or all scenes and may fail for some combinations of objects and viewpoints. Fig. 29 shows the objects obtained by combining the ribbons in Fig. 28 using the above rules.

XI. More Examples

The system has been tested on over 20 images. Some illustrative examples are presented. Fig. 30 shows the image of another scene with curved objects. The edges detected, using a Canny edge-detector, are shown in Fig. 31. The edges are linked into contours, which are then segmented into curves. These curves are grouped on cocurvilinearity, and the resulting curves are displayed in Fig. 32. The handle and cord of the electric iron group as curves, given the scale of the surfaces in the scene. Fig. 33 shows all the symmetry axes proposed, and Fig. 34 shows the selected axes. The axes computed for the base of the iron and the platform of the letter-weighing machine correspond closely to the skew-symmetry axes for these shapes.

The ribbons obtained from these symmetry axes, after recomputing the wrong closures and rejecting the ribbons with wrong or poor closure, are shown in Fig. 35. After the removal of the ribbons forming false T junctions, the remaining ribbons are shown in Fig. 36. The right face of the base of the weighing machine is composed of two ribbons as there is a corner detected along the right boundary of this face; there are, in fact, two slightly offset surfaces composing this face, and there is a thin ridge running between them (see Figs. 30 and 31). There is also a ribbon detected that is a composition of these two ribbons but cannot be seen in the display as it is overlaid by

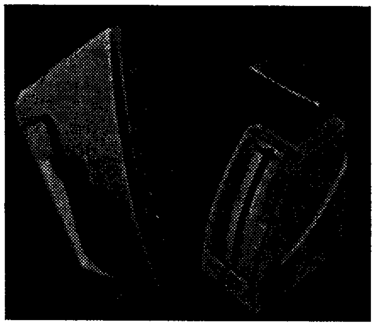


Fig. 30. Image II.

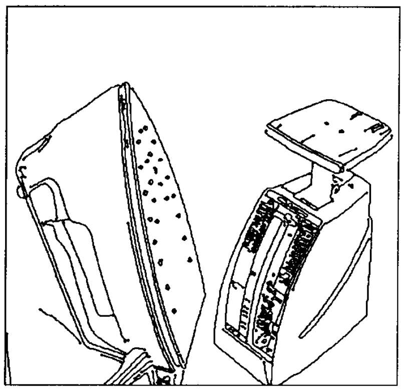


Fig. 31. Edges.

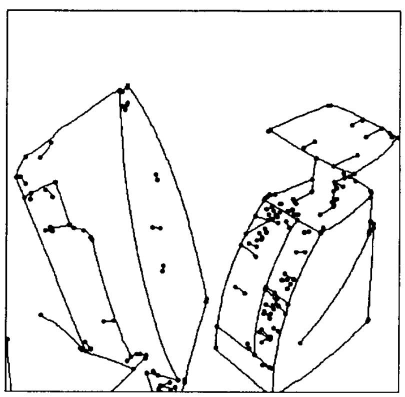


Fig. 32. Curves.

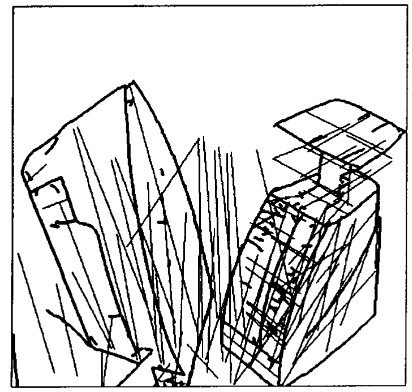


Fig. 33. Hypothesized symmetries.

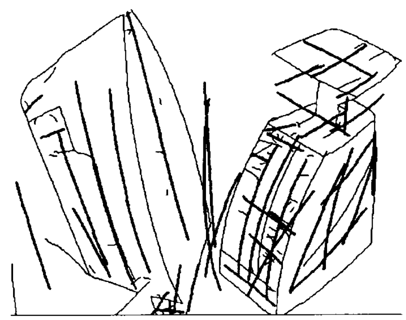


Fig. 34. Selected symmetries.

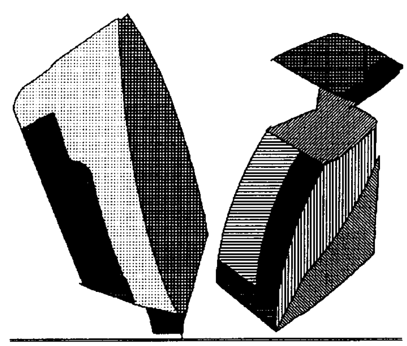


Fig. 35. Ribbons detected.

its component ribbons. Therefore, if at some scale the corner along the right boundary is not detected, then at that scale, we would describe the face by a single surface (as the component ribbons will be removed for forming false T junctions). This effect of the scale of the feature detection on the description of the scene cannot be escaped, and the only effective solution is to maintain descriptions at various scales. For example, if a dense range map of this scene was segmented, the description of the face as one or two surfaces would depend on the ridge being detected by a feature detector at some scale [17]. Fig. 37 shows the objects hypothesized on the basis of the ribbons in Fig. 36.

Fig. 39 shows the edges obtained for image III (Fig. 38). Fig. 40 shows the ribbons detected. Some ribbons, which are compositions of smaller component ribbons, are not shown, such as the ellipse on the bottle and the ribbon composed of the front and top surfaces of the bottle. Fig. 41 shows the ribbons obtained after all interior ribbons (ribbons that are completely contained inside other ribbons but do not share boundary with them) are removed. Although, for this scene, this step removes all ribbons corresponding to surface markings, this type of reasoning is, in general, not valid for all scenes (at least with-

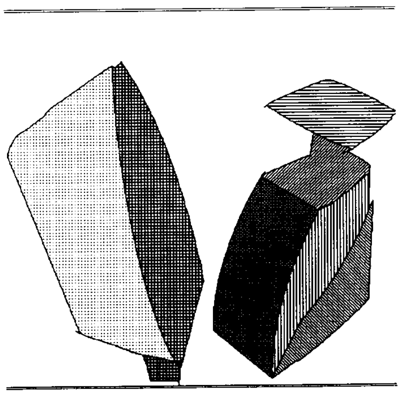


Fig. 36. Surfaces.

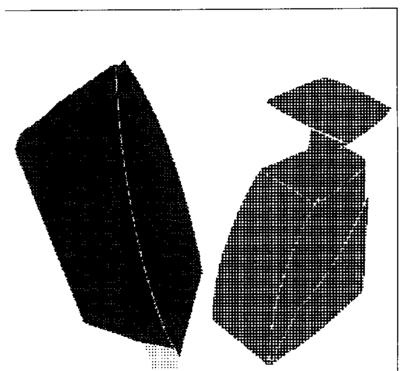


Fig. 37. Objects.

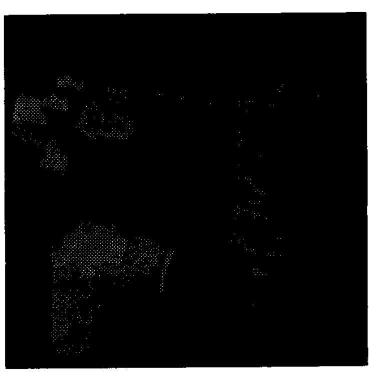


Fig. 38. Images III.

out some 3-D information). Therefore, the ribbon descriptions are best viewed as a hierarchical shape description of the visible surfaces with some ribbons in the hierarchy composed of component ribbons and having associated interior ribbons.

XII. STEREO

In stereo matching, one of the problems is the ambiguity involved in matching edges or intensity windows. Since these features carry little information on which the quality of each match can be independently decided, various optimization

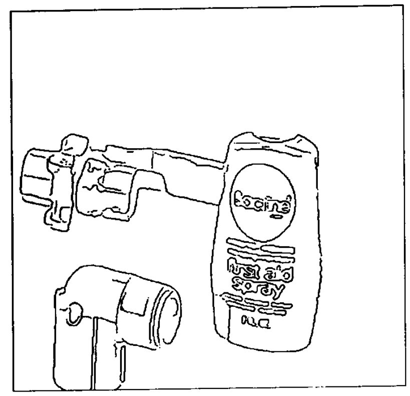


Fig. 39. Edges.

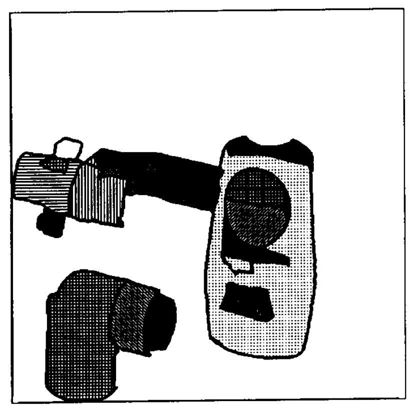


Fig. 40. Surfaces.

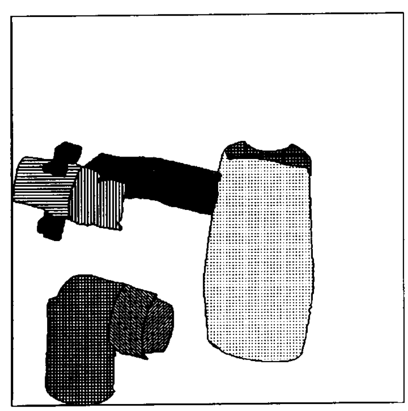
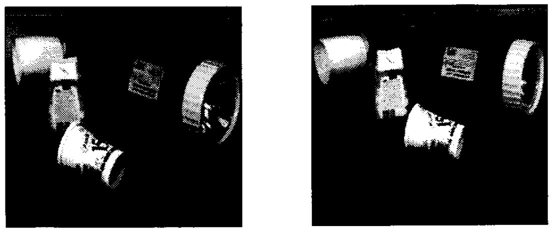


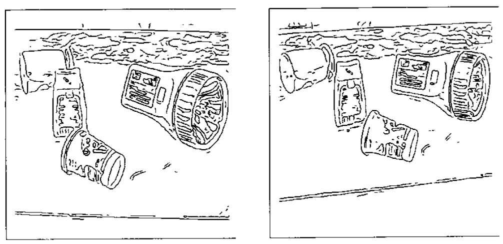
Fig. 41. Interior ribbons removed.

techniques have to be used to obtain the best global match. The number of features matched is extremely large, making both the matching and the optimization computationally expensive. Some of these problems can be alleviated by using more abstract features such as linear segments and contours for the matching [47]. We shall demonstrate the utility of ribbons for correspondence by using them as match primitives in unregistered stereo images (i.e., which have large vertical disparities).



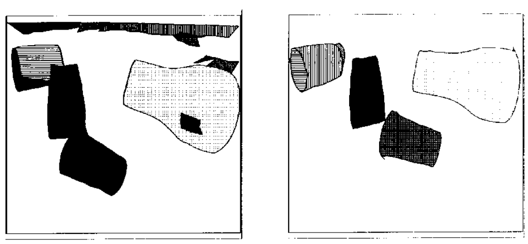
(a) (b)

Fig. 42. Stereo pair I: (a) Right image; (b) left image.



(a) (b)

Fig. 43. Edges detected in stereo pair I: (a) Right image; (b) left image.



(a) (b)

Fig. 44. Ribbons detected in stereo pair I: (a) Right image; (b) left image.

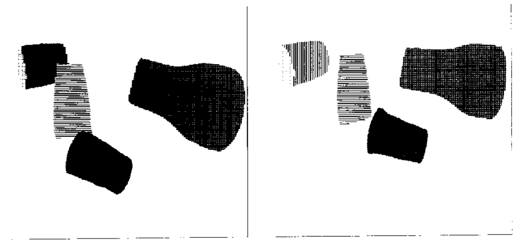
First ribbons are detected in each image of the stereo pair. For each ribbon, an epipolar window is defined in the other image, which accounts for the possible horizontal and vertical disparity. Ribbons are matched on the basis of matches between their axes, the corresponding symmetric curves, and their widths and lengths. The few ambiguous matches are resolved using a quantitative evaluation of the matches and the selection of the best matches by a constraint satisfaction network.

A. Matching Ribbons

A match is evaluated on the basis of the similarity of shapes of the matched ribbons. The factors considered in evaluating a match are the difference in widths of the ribbons, the difference in the lengths and orientations of the corresponding contours and axes, and the amount of epipolar overlap.

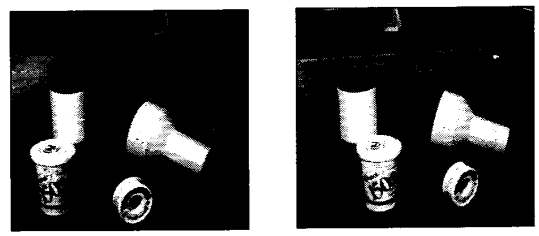
Some results on stereo-matchings on ribbons are shown in Figs. 42 through 49.

The stereo matching has been presented here to demonstrate the utility of collated features for such a task; it is not indented to be a full stereo matching system. Recently, this work has been extended into a comprehensive stereo system [54].



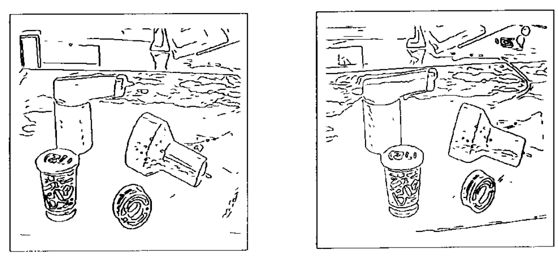
(a) (b)

ig. 45. Matches detected in stereo pair I: (a) Right image; (b) left image.



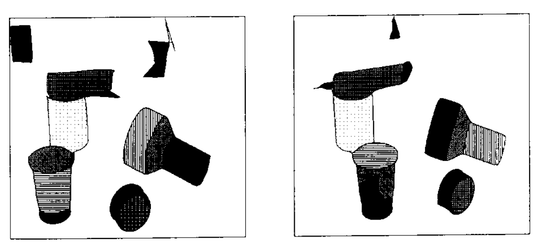
(a) (b)

Fig. 46. Stereo pair II: (a) Right image; (b) left image.



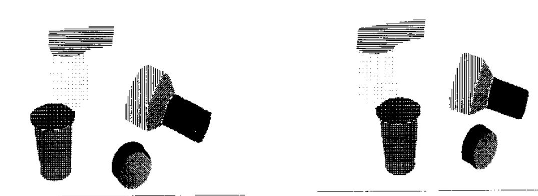
(a) (b)

Fig. 47. Edges detected in stereo pair II: (a) Right image; (b) left image.



(a) (b)

Fig. 48. Ribbons detected in stereo pair II: (a) Right image; (b) left image.



(a) (b)

Fig. 49. Matched ribbons in stereo pair II: (a) Right image; (b) left image.

TABLE II
COMPUTATION TIMES FOR IMAGE I

Process	Complexity	Run Times (min:sec)	Collations
Continuity	$O(n^2)$	4:58	398 curves formed
Proximity	$O(n^2)$	15:28	212 curves formed
Symmetry detection	$O(n^2)$	1:05	163 axes
U-symmetry detection	$O(n^2)$	0:03	11 axes
Symmetry selection	$O(mp)$	1:02	55 axes selected
Ribbon formation	$O(r)$	31:04	59 (55+4) ribbons
structural reasoning	$O(r^2)$	18:33	24 ribbons

TABLE III
COMPUTATION TIMES FOR IMAGE II

Process	Complexity	Run TImes (min:sec)	Collations
Co-curvilinearity	$O(n^2)$	11:37	110 curves formed
Symmetry detection	$O(n^2)$	0:44	85 axes
Symmetry selection	$O(mp)$	0:56	38 selected
Ribbon formation	$O(r)$	26:28	21 ribbons
structural reasoning	$O(r^2)$	13:33	15 ribbons

Collated features are also used to cut down the complexity of the image to model matching tasks [43]. However, the collations previously used [43], [31] apply to limited shapes and/or limited viewpoints. The collated features presented here, especially symmetries and ribbons, will allow the use of these techniques to wider domains.

XIII. Shape from Contour

Monocular interpretations of shape from contour depend on detecting both good contours in the image (i.e., connected contours corresponding to surface boundaries and not texture or markings) and a good axis finding algorithm (for those based on concepts related to skew symmetry). The system presented finds connected contours, which as boundaries of ribbons (in the final segmentation step) correspond to surface boundaries rather than texture. In addition, the axis obtained often corresponds to the skew symmetry axis (always for straight lines). In cases of parallel and mirror symmetries [68], the axis detected could be used directly for the shape analysis.

XIV. Computational Complexity

We present an informal computational complexity analysis of our system, along with run times (on a Symbolic Lisp machine) and number of features. Let there be n curves in the image. Both grouping curves on continuity and on proximity take $O(n^2)$ time. Curve extension takes $O(n)$ time. Computing possible symmetry relationships takes $O(n^2)$ time. If m symmetries are proposed, then symmetry selection takes $O(mp)$ time, where p is the average number of alternate symmetries per symmetry pm. If r symmetries are selected, then ribbon formation takes at most $O(r)$ time. Computing structural relationships between ribbon pairs takes at worst $O(r^2)$ time.

The run times and actual numbers of features are displayed in Tables II and III. In these figures, the time taken for edge detection, linking, and corner detection are not included.

XV. Summary

Perceptual organization is identified as an important component in the processing of images. A technique for performing perceptual organization on real images is presented. Intensity edges are grouped on the basis of their geometrical relationships. The grouping process is recursive and generates a hierarchy of collated features. The collated features encode some of the geometrical information in an image and are used by various visual processes. Thus, one role for perceptual organization for computer vision is identified and implemeted.

Collated features are proposed as the representations computed by the process of perceptual organization applied to primitive image elements. These collations represent structural relationships between the arrangement of their tokens. The structural relationships so represented are characterized in terms of their significance for the shapes in a visual domain and their utility to other visual processes. Specifically, the usefulness of collated features to stereo, the generation of shape descriptions, and object segmentation has been demonstrated.

Currently, the vision system provides only gross structural descriptions, concentrating on features connected with the shape of the surfaces (as opposed to other features like surface markings, which could be significant in the context of correspondence). Future work will concentrate on scale, use of additional cues to aid reasoning, and exploration of other applications.

References

[1] G. J. Agin and T. O. Binford, "Computer description of curved objects," *IEEE Trans. Comput.*, vol. C-25, Apr. 1976.
[2] F. Attneave, "Some informational aspects of visual perception," *Psycholog. Rev.*, vol. 61, pp. 183–193, 1954.
[3] D. H. Ballard, G. E. Hinton, and T. J. Sejnowski. "Parallel visual computation," *Nature*, vol. 306, pp. 21–26, Nov. 1983.
[4] T. O. Binford, "Visual perception by computer," in *Proc. IEEE Conf. Syst. Contr.* (Miami), Dec. 1971.
[5] H. Blum, "Biological shape and visual science (Part 1)," *J. Theoritical Biol.*, vol. 38, pp. 205–287, 1973.
[6] M. Boldt, R. Weiss, and E. Riseman, "Token based extraction of straight lines," *IEEE Syst. Man Cybern*, vol. 19, no. 6, pp. 1581–1594, Dec. 1989.
[7] R. M. Bolle, A. Califano, R. Kjeldsen, and R. W. Taylor, "Visual recognition using concurrent and layered parameter networks," in *Proc. IEEE Conf. Comput. Vision Patt. Recogn.* (San Diego, CA), June 1989.
[8] M. Brady, "Criteria for representation of shapes," in *Human and Machine Vision* (J. Beck, B. Hope, and A. Rosenfeld, Eds.). New York: Academic, 1983, pp. 39–84.
[9] M. Brady and H. Asada, "Smoothed local symmetries and their implementation," *Int. J. Robotics Res.*, vol. 3, pp. 36–61, Fall 1984.
[10] R. A. Brooks, "Goal-directed edge linking and ribbon finding," in *Proc. DARPA Image Understanding Workshop*, Apr. 1979, pp. 72–78.
[11] ——, "Model-based three-dimensional interpretation of two-dimensional images," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-5, no. 2, pp. 140–150, Mar. 1983.
[12] J. F. Canny, "A computational approach to edge detection," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.
[13] J. H. Connell and M. Brady, "Generating and generalizing models of visual objects," Tech. Rep. AIM-823, Mass. Inst. Technol., July 1985.
[14] J. Dolan and R. Weiss, "Perceptual grouping of curved lines," in *Proc. DARPA Image Understanding Workshop* (Palo Alto, CA), 1989, pp. 1135–1145.
[15] S. E. Fahlman and G. E. Hinton, "Connectionist architectures for artificial intelligence," *IEEE Comput.*, pp. 100–109, Jan. 1987.
[16] S. E. Fahlman, G. E. Hinton, and T. J. Sejnowski, "Massively parallel architectures for AI: NETL, Thistle, and Boltzman machines," in *Proc. Nat. Conf. AI* (Menlo Park, CA), 1983.
[17] T. -J. Fan, "Describing and recognizing 3-D objects using surface properties," Ph.D. Thesis, Tech. Rep. IRIS-237, Inst. Robotics Intell. Syst., Univ. Southern Calif., Aug. 1988.
[18] J. A. Feldman and D. H. Ballard, "Connectionist models and their properties," *Cognitive Sci.*, pp. 205–254, 1982.
[19] M. A. Fischler and R. C. Bolles, "Perceptual organization and curve partioning," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-8, no.

1, pp. 100–105, Jan. 1986.

[20] W. R. Garner, *The Processing of Information and Structure*. Potomac, MD: Erlbaum, 1974.

[21] W. R. Garner and D. E. Clement, "Goodness of pattern and pattern uncertainity," *J. Verbal Learning Verbal Behavior*, vol. 2, pp. 446–452, 1963.

[22] S. L. Gazit and G. Medioni, "Multi-scale contour matching in a motion sequence," in *Proc. DARPA Image Understanding Workshop*, May 1989.

[23] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-6, no. 6, pp. 721–741, Nov. 1984.

[24] W. E. L. Grimson, "The combinatorics of heuristic search termination for object recognition in cluttered environment," in *Proc. First Euro. Conf. Comput. Vision*, 1990, pp. 552–556.

[25] J. E. Hochberg and E. McAllister, "A quantitative approach to figural 'goodness,'" *J. Experimental Psychol.*, vol. 46, pp. 361–364, 1953.

[26] J. J. Hopfield, "Neurons with graded response have collective computational properties like those of two-state neurons," *Proc. Nat. Acad. Sci.* vol. 81, pp. 3088–3092, May 1984.

[27] J. J. Hopfield and D. W. Tank, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Nat. Acad. Sci.*, vol. 79, pp. 2554–2558, Apr. 1982.

[28] ——, "'Neural' computation of descisions in optimization problems," *Biol. Cybern.*, vol. 52, pp. 141–152, 1985.

[29] ——, "Computing with neural circuits: A model," *Sci.*, vol. 233, pp. 625–633, 1986.

[30] R. A. Hummel and S. W. Zucker, "On the foundations of relaxation labeling process," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-5, no. 3, pp. 267–287, May 1983.

[31] D. J. Jacobs, "GROPER: A grouping based recognition system for two dimensional objects," in *Proc. IEEE Comput. Soc. Workshop Comput. Vision* (Miami Beach, FL), Dec. 1987.

[32] B. Julesz, "Figure and ground perception in briefly presented isodipole textures," in *Perceptual Organization* (Kubovy and Pomerantz, Eds.). Hillsdale, NJ: Lawrence Erlbaum, 1981, pp. 27–54.

[33] T. Kanade, "Recovery of the three-dimensional shape of an object from a single view," Tech. Rep. CMU-CS-79-153, Carnegie-Mellon Univ., Pittsburgh, PA, Oct. 1979.

[34] T. Kanade and J. R. Kender, "Mapping image properties into shape constraints: Skewed symmetry, affine-transformable patterns, and the shape-from-texture paradigm," in *Human and Machine Vision* (J. Beck, B. Hope, and A. Rosenfeld, Eds.). New York: Academic, 1983, pp. 237–268.

[35] G. Kanisza, *Organization in Vision*. New York: Praeger, 1979.

[36] D. Katz, *Gestalt Psychology: Its Nature and Significance*. New York: Ronald, 1950.

[37] R. E. Kelly, P. R. M. McConnell, and S. J. Mildenberger, "The gestalt photomapping system," *J. Photogrammetric Eng. Remote Sensing*, 1977.

[38] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Sci.*, vol. 220, pp. 671–680, 1983.

[39] Kubovy and Pomerantz, Eds., *Perceptual Organization*. Hillsdale, NJ: Lawrence Erlbaum, 1981.

[40] K. I. Laws, "Textured image segmentation," Ph.D. dissertation, Univ. Southern Calif., 1980.

[41] E. L. J. Leewburg, "A perceptual coding language for visual and auditory patterns," *Amer. J. Psychol.*, vol. 84, pp. 307–350, 1971.

[42] H. S. Lim and T. O. Binford, "Stereo correspondence: A hierarchical approach," in *Proc. DARPA Image Understanding Workshop* (Los Angeles), Feb. 1987, pp. 234–241.

[43] D. G. Lowe, *Perceptual Organization and Visual Recognition*. Hingham, MA: Kulwer Academic, 1985.

[44] D. G. Lowe and T. O. Binford, "Perceptual organization as a basis for visual recognition," in *Proc. AAAI-83* (Washington, DC), Aug. 1983.

[45] J. Malik, "Interpreting line drawings of curved objects," *Int. J. Comput. Vision*, vol. 1, no. 1, pp. 73–103, 1987.

[46] D. Marr, *Vision*. San Francisco: W. H. Freeman, 1982.

[47] G. Medioni and R. Nevatia, "Segment-based stereo matching," *Comput. Vision Graphics Image Processing*, vol. 31, pp. 2–18, 1985.

[48] R. Mohan, "Application of neural constraint satisfaction networks to vision," in *Proc. Int. Joint Conf. Neural Networks* (San Diego), 1989.

[49] ——, "Perceptual organization for computer vision," Ph.D. Thesis, IRIS Rep., Univ. Southern Calif., 1989.

[50] R. Mohan and R. Nevatia, "Segmentation and description based on perceptual organization," in *Proc. IEEE Conf. Comput. Vision Patt. Recogn.* (San Diego, CA), June 1989.

[51] ——, "Using perceptual organization to extract 3-D structures," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 11, no. 10, Oct. 1989.

[52] L. R. Nackman and S. M. Pizer, "Three-dimensional shape description using the symmetric axis transform," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-7, pp. 187–201, Mar. 1985.

[53] R. Nevatia, *Computer Analysis of Scenes of 3-D Curved Objects*. Basel, Switzerland: Birkhauser-Verlag, 1976.

[54] R. Nevatia, K. Price, and G. Medioni, "USC image understanding research 1989–1990," in *Proc. DARPA Image Understanding Workshop* (Pittsburgh), Sept. 1990.

[55] R. Nevatia and T. O. Binford, "Description and recognition of complex-curved objects," *Artificial Intell.*, vol. 8, pp. 77–98, 1977.

[56] S. E. Palmer, "The psychology of perceptual organization: A transformational approach," in *Human and Machine Vision* (J. Beck, B. Hope, and A. Rosenfeld, Ed.). New York: Academic, 1983, pp. 269–339.

[57] J. Ponce, D. Chelberg, and W. Mann, "Analytical properties of generalized cylinders and their projections," in *Proc. DARPA Image Understanding Workshop* (Los Angeles, CA), Feb. 1987.

[58] L. Quan, R. Mohr, and E. Thirion, "Generating the initial hypothesis using perspective invariants for a 2D image and 3D model matching." in *Proc. Int. Conf. Comput. Vision* (Tarpon Springs, FL), Dec. 1988.

[59] K. Rao, "Shape description from sparse and imperfect data," Ph.D. Thesis, Tech. Rep. IRIS-250, Inst. Robotics Intell. Syst., Univ. Southern Calif., Dec. 1988.

[60] G. Reynolds and J. R. Beveridge, "Searching for geometric structure in images of natural scenes," in *Proc. DARPA Image Understanding Workshop* (Los Angeles, CA), Feb. 1987.

[61] A. Rosenfeld, "Axial representation of shape," *Comput. Vision Graphics Image Processing*, vol. 33, pp. 156–173, 1986.

[62] D. E. Rumelhart *et al.*, *Parallel Distributed Processing: Explorations in the Microstructures of Computing*. Cambridge, MA: MIT Press, 1986.

[63] P. Saint-Marc and G. Medioni, "Adaptive smoothing for feature extraction," in *Proc. DARPA Image Understanding Workshop* (Boston, MA), Apr. 1988.

[64] K. A. Stevens, "Computation of locally parallel structure," *Biol. Cybern.*, vol. 29, pp. 19–28, 1981.

[65] K. A. Stevens and A. Brookes, "Detecting structures by symbolic constructions on tokens," *Comput. Vision Graphics Image Processing*, vol. 37, pp. 238–260, 1987.

[66] A. Triesman, "Perceptual grouping and attention in visual search for features and objects," *J. Experimental Psychol.: Human Perception Performance*, vol. 8, no. 2, pp. 194–214, 1982.

[67] M. Tuceryan and N. Ahuja, "Perceptual segmentation of nonhomogeneous dot patterns," in *Proc. Comput. Vision Patt. Recogn. Conf.* (Washington, DC), June 1983, pp. 47–52.

[68] F. Ulupinar and R. Nevatia, "Using symmetries for analysis of shape from contour," in *Proc. Int. Conf. Comput. Vision* (Tampa, FL), Dec. 1988.

[69] D. L. Waltz, "Understanding line drawings in scenes with shadows," in *The Psychology of Computer Vision* (P. H. Winston, Ed.). New York: McGraw-Hill, 1975, pp. 19–91.

[70] M. Wertheimer, "Principles of perceptual organization," in *Readings in Perception* (D. Beardslee and M. Wertheimer, Eds.). Princeton, NJ: Princeton University Press, 1958, pp. 115–135.

[71] A. Witkin, "Scale-space filtering," in *Proc. Seventh Int. Joint Conf. Artificial Intell.* (Kalsruhe, West Germany), 1983.

[72] A. P. Witkin and J. M. Tenenbaum, "On the role of structure in vision," in *Human and Machine Vision* (J. Beck, B. Hope, and A. Rosenfeld, Eds.). New York: Academic, 1983, pp. 481–543.

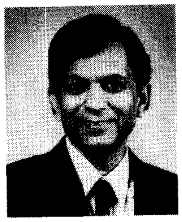
[73] S. W. Zucker, "Computational and psychophysical experiments in grouping: Early orientation selection," in *Human and Machine Vision* (J. Beck, B. Hope, and A. Rosenfeld, Eds.). New York: Academic, 1983, pp. 545–567.

[74] ——, "The diversity of perceptual grouping," in *Vision, Brain, and Cooperative Computation* (M. A. Arbib and A. R. Hanson, Eds.). Cambridge, MA: MIT Press, 1987, pp. 231–262.

Rakesh Mohan (S'84–M'90) received the B.Tech degree in computer science from the Indian Institute of Technology, Kanpur, India, in 1983 and the M.S. and the Ph.D degrees in computer science from the University of Southern California, Los Angeles, in 1989.

He is currently a Research Staff Member at the IBM Thomas J. Watson Laboratory, Yorktown Heights, NY. His research interests include computer vision, robotics, and neural networks.

Dr. Mohan is a co-editor of the book *Progress in Neural Networks: Neural Networks in Vision*.



Ramakant Nevatia (SM'86–F'91) received the B.S. degree from the University of Bombay, India, and the M.S. and Ph.D. degrees from Stanford University, Stanford, CA, all in electrical engineering.

He has been with the University of Southern California, Los Angeles, since 1975, where he is currently Professor of Computer Science and Electrical Engineering and Director of the Institute for Robotics and Intelligent Systems. He spent the 1981–1982 academic year at Stanford University as a Visiting Professor. His research interests include computer vision, artificial intelligence, and robotics.

Dr. Nevatia is a member of the Association for Computing Machinery. He has authored two books: *Machine Perception* and *Computer Analysis of 3-D Curved Objects*. He is an Associate Editor of the journals *Pattern Recognition* and *Computer Vision, Graphics, and Image Processing* and was a Technical Editor in the areas of robot vision and inspection systems for the IEEE JOURNAL OF ROBOTICS AND AUTOMATION.