Robotica (1987) volume 5, pp 117–127

Automatic generation of salient features for the recognition of partially occluded parts*

T.N. Mudge, J.L. Turney and R.A. Volz

Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, Michigan 48109-1109 (U.S.A.)

(Received: May 31, 1986)

SUMMARY

A method for solving the recognition of partially occluded parts is presented. It is based on the automatic generation of features from a set of primitive features which are configurations of pairs of fixed length segments of boundary edges of the parts. The procedure that creates the recognition features assigns a number in the range (0, 1] that indicates the importance of the feature in the recognition strategy. This number is referred to as the feature's saliency. The method assumes that the parts that can occur in a scene come from a known set of parts. An example illustrates how automatically generated features can be used to count the number of identical parts in a heap.

Keywords: Automatic; Recognition; Occluded; Generation; Features

I. INTRODUCTION

In many practical applications of computer vision the basic vision task is that of recognizing one or more parts in a digitized image where the parts may be partially occluded. The partial occlusion normally results from allowing the parts to overlap one another. This overlapping greatly complicates the recognition problem. This problem of overlapping parts is sometimes named for a paradigm, the *bin of parts problem,* which involves recognizing parts piled in a bin, a common way in which parts are presented for batch assembly. The bin of parts problem has been described as "the most difficult problem in automatic assembly." A solution is said to be worth "tens of millions of dollars a year in the U.S."[1] The bin of parts problem is common to many industrial tasks such as part sorting, part retrieval, and part assembly, and, as yet, there is no satisfactory solution to this problem.

Partial occlusion is also found in images where parts are obscured by dirt, where parts are defective, or where parts are partially outside an image. Images with these characteristics present basically the same problem to recognition as occlusion from overlap: only some areas of the parts are exposed, and the parts must be recognized from these exposed areas. In this discussion the general recognition problem will be called the *partially occluded parts* problem – the POP problem for short. Overlapping, obscured, defective, and incomplete views of parts are all instances of the generic POP problem.

This paper presents a method that recognizes partially occluded parts by using a set of features that are automatically generated from a set of primitive features. The procedure that creates the features assigns a number in the range (0, 1] that indicates the importance of each feature in the recognition strategy. This number is referred to as a feature's *saliency*. Saliency, in effect, characterizes the relative importance of certain aspects of a feature's shape and is used to determine the order in which a search for recognition is made. The method assumes that the set of parts that may appear in an image is known *a priori.* This *a priori* knowledge is what allows us to calculate the saliency during an off-line training step. The method is restricted to *2-dimensional* parts. For our purposes, a part is 2-dimensional if two of the dimensions of the part are much larger than the third. A solution to the 2-dimensional POP problem is nearly as useful as a solution to the general POP problem; applications involving stamped, cast, and forged flat metal parts are common in industry.

Figure 1 shows a bin of parts. In this example the parts are similar but can appear in two distinct stable positions. The application is to count the number of parts

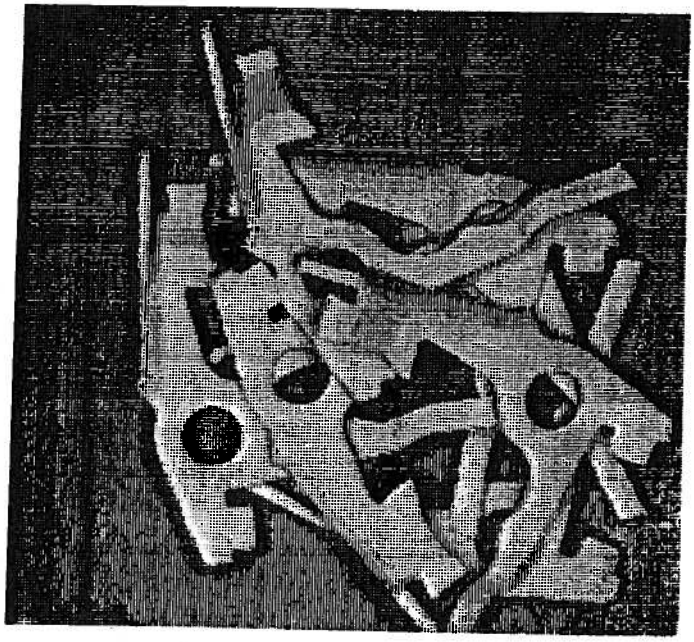

Fig. 1. A bin of parts.

* This work was supported in part by a grant from the US Army Research Office under contract DAAG29-84-K-0070.

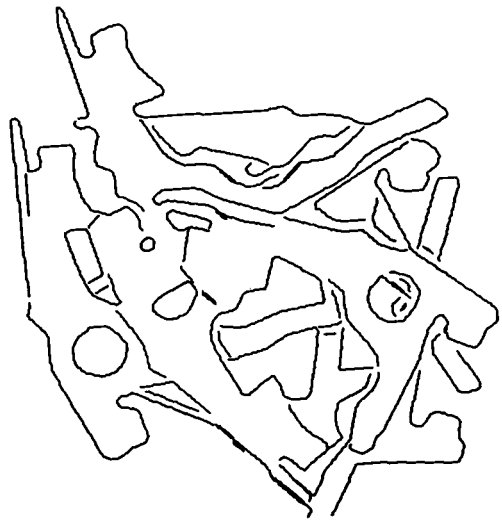

Fig. 2. Boundaries from the bin of parts.

in the bin. The figure shows the grey level image of the parts. It is a 256×256 image of 8 bit pixels. We assume that some form of preprocessing is performed on the image to extract pixel wide edge boundaries (see Figure 2). The off-line training step acquires shape information either by viewing each part alone in each of its stable positions, or directly from a solid model of each part. (For industrial parts the latter may be available from a CAD database.) The shape information is the set of pixel wide edge boundaries of the part. Figure 3 shows the boundary of the part that appears in the bin of parts shown in Fig. 1.

The method under discussion provides solutions to many problems in industrial vision that current systems cannot handle. The method can recognize 2-dimensional parts under any of the following conditions:

1. The parts may be located at any spatial position, or under any rotation about the viewing axis.
2. The parts may touch, overlap, lie partially outside the image, be dirty or have defects.
3. The parts may be reflective and viewed under poor lighting conditions.
4. The parts may be viewed with any scale within a wide range.

The primary focus of this paper is the concept of a salient feature. It does not discuss the last two recognition problems. They are discussed in detail in ref. 2, which also compares the performance of the method to that of other popular approaches.

This paper is organized as follows. In the following section we present arguments for using pairs of fixed length overlapping boundary segments, termed configurations, as primitive features. An efficient method is presented for determining matches between the configuration-segments used to recognize parts and segments in the image. Section III presents a method for automatically generating salient features from the primitive features. The section begins by formalizing the concept of saliency and goes on to present a training method for the automatic generation of salient configurations. Section IV presents a strategy for solving the POP problem that makes use of the salient features. Section V concludes the paper.

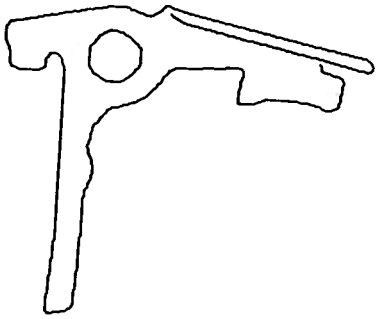

Fig. 3. Boundary of a part.

II. PRIMITIVE FEATURES

The choice of pairs of fixed length overlapping boundary segments as primitive features can be better appreciated if features used in other recognition methods are briefly reviewed first.

1. *Review*

Merlin and Farber[3] and Ballard[4] use *edge points* as their primitive features. Edge points are a good representation in the presence of occlusion, because, short of total occlusion, some edge points of a part-model will always appear in the image. Unfortunately edge points have the disadvantage that they are indistinguishable from one another: edge points can belong to any boundary of any part in the image. For this reason, as is shown in ref. 2, the generalized Hough transforms of Merlin and Farber, and Ballard often find incorrect locations for parts. The edge points of the part may correlate better with edge points that belong to other parts, or to the wrong edge points on the same part, than to the correct set of edge points.

Rutkowski[5] uses edge points with associated *probabilistic labeling vectors and relations between edge points* as primitive features. A probabilistic labeling vector $\mathbf{p}^i$ is associated with each edge point i of the image. Element p^i_j represents the probability that image point i can be labeled as edge point j of the part. Rutkowski uses the labeling vectors and spatial relations between edge points as input to a relaxation algorithm. The method appears unnecessarily complex and results in undesirably long times.

Kelly et al.[6] and Jacobsen and Wechsler[7] use *gray level regions* as primitive features. A gray level region is no more than a contiguous set of gray valued pixels. By correlating certain regions with an image, one can estimate the location and identity of parts in the image. If we adopt the approach in ref. 6, region correlation requires $O(n^2m^5)$ operations, when images have $n \times n$ pixels, regions have $m \times m$ pixels, and m^3 different views of each region are used. Region correlation is sensitive to occlusions because it is unreliable when based on small regions. In a POP image the visible regions of a part will generally be small, and, therefore, regions are inadequate features for the POP problem.

Primitive features based on *axial representations*, such as the symmetric axis transform (SAT) of Blum and Nagel[8] and the smoothed local symmetries (SLS) of

Brady and Asada,[9] are simple features to identify, but they will only match corresponding representations of images when the images are essentially free of occlusion. The axial representation of a part can be dramatically changed by relatively minor subtractions from the part's shape (such as might occur from occlusion). This makes finding a match with an axial representation of the part difficult.

Bolles and Cain,[10] Berman et al.,[11] Tropf,[12] Koch and Kashyap,[13] and Stockman et al.[14] use *special features*, such as corners and holes, as primitive features. Special features have the advantage that they are easy to locate and are thus useful for quick recognition. However, special features occur infrequently on the contours of typical parts, making them vulnerable to occlusion. In general, algorithms that rely solely on special features often have difficulty finding enough features in the image for reliable recognition. In addition, algorithms based on special features are problem specific. Without an automatic way to select special interesting features from a set of parts, it is necessary to redesign an algorithm for each problem domain.

Ayache and Faugeras[15] use *line segments* that form the sides of a polygon approximation of the boundary of a part as primitive features. Line segments are more frequent in typical images than special features, but are distinguishable from each other only by their length. When recognizing a part, Ayache and Faugeras compare only the longer segments from the part-model to the segments in the image. This dramatically reduces the number of comparisons from the number required if they had compared line segments of arbitrary length. Unfortunately, in POP images the longer line segments are more likely to be occluded, and, thus, are generally not a good choice of features.

Our brief survey suggests that probabilistic labeling vectors, regions, axial representations, special features, and line segments, are not the most suitable features for use in solving the POP problem. By comparison, edge points appear better suited to the problem. Their main shortcoming is that they have no local structure to differentiate one edge point from another. This can be overcome by using sequences of linked points, or *segments*. Moreover, to maximize the number of visible segments in the presence of occlusion, segments should be overlapped. Therefore, we have chosen fixed length overlapping segments of the boundary as components in our primitive features. The fixed length is a design parameter and depends on the set of known parts. If the length is too short, the segments will be indistinguishable. On the other hand, if the length is too long, the segments are less likely to be visible, although longer segments are implicitly compared when shorter fixed length segments that cover the longer segments are compared. The fixed segment length should be related to the curvature of the parts: if there is frequent occurrences of high curvature, segments should be short. Figure 3 shows the boundary of one of the stable positions of the part that appears in the bin of Fig. 1. Figure 4 repeats Fig. 2 with the visible boundary segments of one instance of the part highlighted. Figure 5 shows overlapped segments AA′, BB′, and CC′.

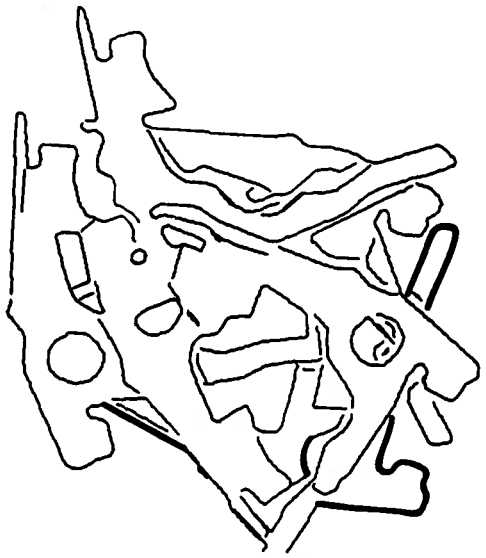

Fig. 4. The visible segments of a part in the bin.

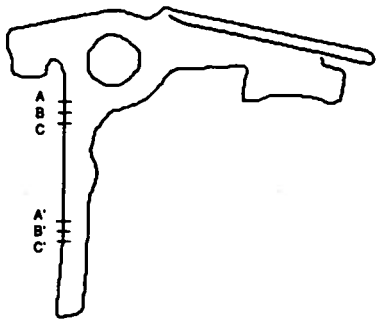

Fig. 5. Overlapping segments.

The set of overlapping segments are obtained from all the boundaries from all of the part's stable positions. Since the part is 2-dimensional, we make the assumption that it will always appear in a quasi-stable position, that is to say, tilted little from one of its true stable positions, even when it is in a pile with other parts. Finally, to minimize the possibility that a segment, resulting from the random alignment of two or more boundary segments from different parts in an image, can match one that may be used for recognition, we have chosen to use *configurations* of segments as primitive features. Figure 6 shows a configuration. It is simply two fixed length segments in a fixed relative position. The relative position of two segments can be defined by the angle

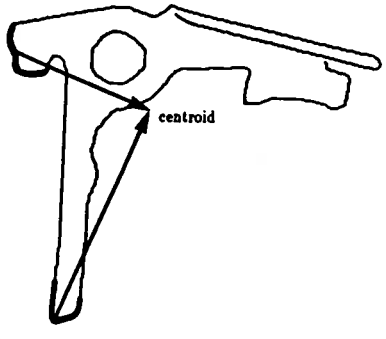

Fig. 6. A configuration of segments.

between the vectors from the mid-points of the segments to the centroid of the part's boundary from which the segments are taken. (This definition implicitly fixes the length of vectors and the angles they make with the tangents at the mid-point of the segments.) The notion of a configuration could be extended to allow an arbitrary number of components or segments. This extension is similar to Bolles and Cain's *local feature focus*, which is a configuration of special features.[10] The case of configurations with other than two components, however, will not be considered further.

Configurations form our primitive features. However, using them indiscriminately for recognition would not result in an efficient algorithm, since each part has a large number of configurations which are nearly indistinguishable. To counter this we have adopted a measure for the distinctiveness of a configuration which we term the *saliency* of the configuration. Informally, the saliency of a configuration is the inverse of the frequency of occurrence of the configuration in the set of known parts. The idea behind this notion is that the more often a configuration is found in the set of parts, the less important the configuration is in distinguishing a part and its pose. An efficient recognition strategy begins by trying to identify the most salient configurations first. The concept of saliency must be modified slightly when noise is taken into account, as we shall see. However, before exploring saliency more fully, we first present an efficient method for determining matches between configuration-segments used to recognize parts and segments in the image.

2. Segment matching

To avoid confusion we shall refer to configurations generated during training as *model* configurations to distinguish them from configurations in the image. Similarly, we shall refer to model segments or sometimes model configuration-segments when we wish to emphasize that they are components of a configuration.

The information needed to recognize a part includes a dual representation for the model configuration-segments, a straightforward *Cartesian* representation and a $\theta - a$ representation (see Figure 7). The $\theta - a$ representation is a parameterization of the slope angle, θ, of the part's boundary by its arclength, a, where arclength is measured from an arbitrary starting point on the boundary. The slope angle can be represented as a function of arclength, $\theta(a)$. The $\theta - a$ representation allows us to compare model segments with image segments that are flipped, that are scaled, and that occur in images with contrast reversals, in each case more efficiently than in a Cartesian representation (for more on these cases see ref. 2). The $\theta - a$ representation does not, however, preserve 2-dimensional distances between segments. Therefore, to compare configurations it is necessary to compare the segments of both configurations individually in $\theta - a$ space and then to check the relative poses between segments in Cartesian space. We will assume that two configurations matched if their

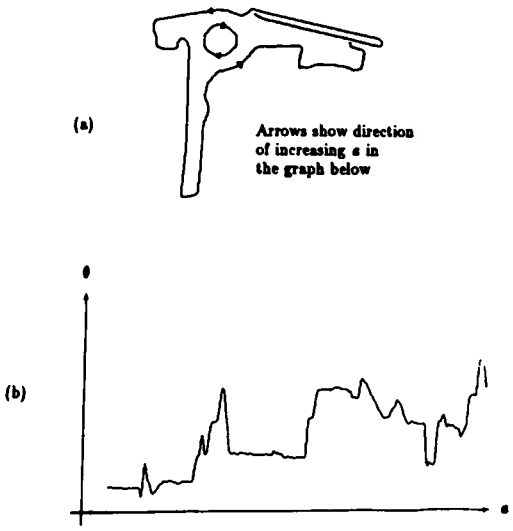

Fig. 7. A part and its $\theta - a$ representation.

segments have the same relative pose in Cartesian space and if the corresponding segments of the configurations match in $\theta - a$ space.

Comparing configurations. In a Cartesian representation, segments are compared by fitting a segment of the part to a segment of the image. Fitting involves three parameters: x and y components of translation, and a relative rotation angle, ϕ. In the $\theta - a$ representation the segments can be fit with one parameter, the relative orientation, Θ.

To compare a model segment with an image segment, we select the sum of the squares of the differences between corresponding slope angles as a measure of the closeness of the fit. The centers of the segments are aligned and the θ values of the model segment, $\theta_M(a_i)$ for $i = -n, \ldots, n$, are least squares fit to the corresponding θ values of the image segment, $\theta_I(a_i)$ for $i = -n, \ldots, n$. We assume that both segments have been sampled at equal arclengths at n points on either side of their centers. The fit parameter, Θ, is chosen to minimize the following sum of squares,

$$\frac{1}{2n+1}\sum_{i=-n}^{n}(\theta_M(a_i) - \theta_I(a_i) - \Theta)^2.$$

The minimum occurs when

$$\Theta = \frac{1}{2n+1}\sum_{i=-n}^{n}(\theta_M(a_i) - \theta_I(a_1)) = \bar{\theta}_M - \bar{\theta}_I,$$

in other words, when Θ is simply the difference between the mean tangent angles of the two segments. The minimum residue

$$R = \sqrt{\frac{1}{2n+1}\sum_{i=-n}^{n}(\theta_M(a_i) - \bar{\theta}_M - (\theta_I(a_i) - \bar{\theta}_I))^2} \quad (1)$$

can be used as a measure of the similarity of the segments, and is used to decide whether the segments match. Equation (1) is all that must be calculated to compare segments once θ and a are determined. In practice we assume that two segments match if R is less

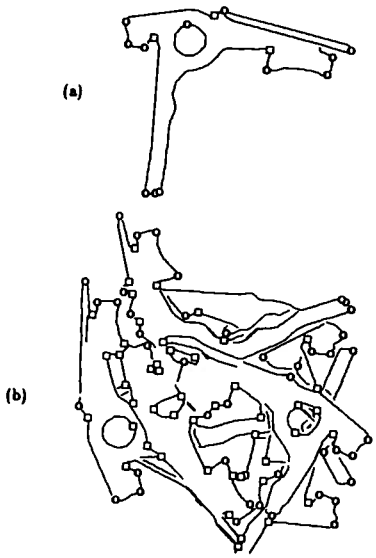

Fig. 8. Critical points of a part and a POP image.

than a fixed threshold, D. The value of D is chosen to reflect the noise anticipated in the images under consideration. We assume that two configurations match when

1. the relative poses of the segments of the configurations in cartesian space are equal and,
2. the segments of one configuration individually match corresponding segments of the second configurations in $\theta - a$ space, i.e., are within a tolerance D of each other.

Critical points. Critical points in a boundary, if they exist, can be used to further improve the efficiency of comparison. We define critical points as the maxima and minima of the curvature of the boundary, i.e., $d\theta(a)/da$ (see ref. 16), that have a curvature value above a fixed threshold. If a model segment contains critical points, as is often the case, it need only be compared to image segments that contain critical points at the same relative positions, thus substantially reducing the number of comparisons needed to locate matches. The location of critical points in the boundary is readily obtained by applying a 1-dimensional edge detector to the function $\theta(a)$. Figure 8(a) shows the critical points of a part; curvature maxima are shown as circles and curvature minima are shown as squares. Figure 8(b) shows the critical points of the boundary in a POP image. Note the correspondence of critical points.

Finally, we note that it is not necessary to store all the segments of both representations; it is sufficient to store the Cartesian boundary and $\theta(a)$ as a linked list of edge points from which segments may be taken during run-time.

III. AUTOMATIC GENERATION OF SALIENT RECOGNITION FEATURES

1. *Saliency*

The concept of saliency, a measure of the importance of a feature in identifying the part, is central to our recognition algorithm. An earlier version of this concept was presented in ref. 17. The saliency of a configuration is learned during off-line training from the boundaries of the complete set of parts that may appear in the image. To simplify the presentation of the concept we begin by considering the case where there is no noise in the image. We then show this is a special case of saliency in the presence of noise.

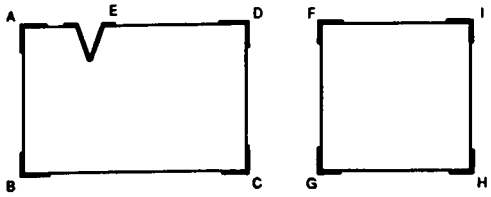

Fig. 9. Parts without noise.

Saliency without noise. If parts are viewed without noise and all parts appear with equal likelihood in an image, we define the saliency of a configuration of segments to be the inverse of the frequency with which identical configurations appear in the set of parts. For example, assume that the notched rectangle and the square shown in Figure 9 are the set of parts that may appear in an image, and that both have equal probability of appearing. The configuration of corner A and corner B has a saliency of $1/(2+4)$ or $\frac{1}{6}$, since identical configurations appear twice in the rectangle and four times in the square. Figure 10 shows how saliency is computed for the configuration of corners A and B. The dashed outlines indicate the six alignments of the set of parts that yield matches. The notation X-Y means that segment X from one of the parts is matched with segment Y of the rectangle. Note that, in effect, both parts are moved around to find matches with the

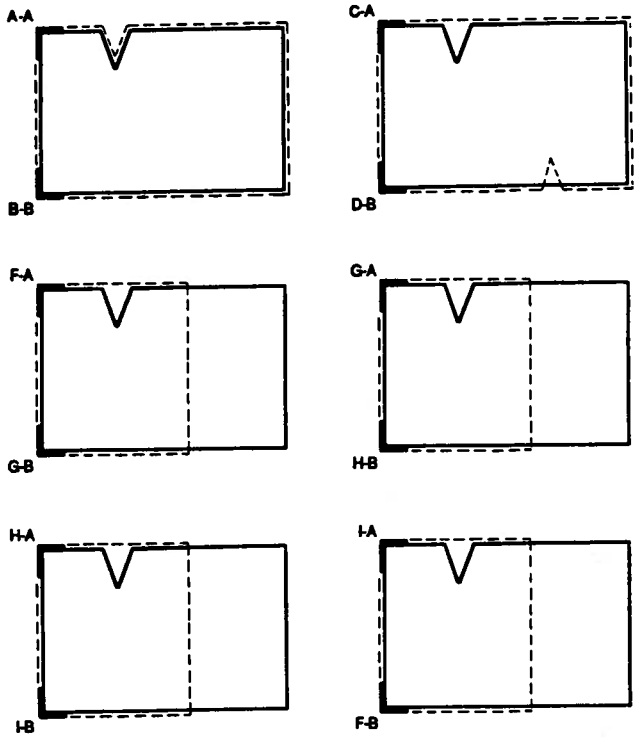

Fig. 10. Computing saliency.

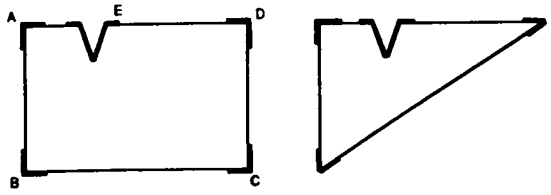

Fig. 11. Notched rectangle and triangle without noise.

configuration A and B. We could just as easily have imagined the matching process as moving the configuration while the parts were held fixed. Continuing our example, we see that the configuration of corner A and corner C has a saliency of $\frac{1}{2}$ since identical configurations appear twice in the rectangle. Finally, the configuration of corner A and the notch E has a saliency of 1 since this configuration appears only once in all the parts: it uniquely characterizes the rectangle and its pose in the image. If this configuration appears in the image, the pose of the rectangle is known with probability one, barring accidental alignments (see below). The calculation of saliency depends on knowing the set of all parts that may appear in an image, consequently features that make use of saliency implicitly incorporate comparative information about the particular part set.

Clearly, saliency is highly dependent on the set of parts. To illustrate this consider a set that contains only the notched rectangle of Figure 9 and a similarly notched triangle (see Figure 11). Reexamining the saliency of the configurations in the rectangle reveal that the configuration of A and B now has a saliency of $\frac{1}{2}$; the configuration of A and C still has a saliency of $\frac{1}{2}$; but the configuration of A and the notch E now has a saliency of $\frac{1}{2}$ and thus no longer uniquely identifies the pose of the rectangle. However, the pose of the rectangle is still uniquely identified by C and E.

The saliency of a configuration is in some sense indivisible; it cannot, in general, be determined from the frequency of occurrence of its component segments. For example, in Figure 9 the saliency of the configuration formed by corners A and B is $\frac{1}{6}$, and the saliency of A and C is $\frac{1}{2}$, while the inverse of the frequency of the individual segments A, B, and C are all $\frac{1}{8}$.

Saliency with noise. We now turn to consider noisy images. If noise is present in an image the saliency of a configuration may be defined by generalizing the probabilistic viewpoint introduced informally above. In the following, we will adapt the simple Bayesian argument of ref. 18 to formalize the concept of saliency for parts when noise is present in the image. First some notation. Let $B_p(x, y, \theta)$ describe the boundary of part p from the set of known parts. The parameters x and y are the coordinates of the centroid of the boundary and θ is the orientation of the boundary about the centroid—different values of x, y and θ correspond to different poses. In practice x, y, and θ are restricted to a finite set of values as a result of digitization. Thus, it is more accurate to represent $B_p(x, y, \theta)$ as $B_p(x_i, y_j, \theta_k)$ where i, j, and k are indices for the finite set of values to which x, y, and θ are restricted. Assume that $B_p(x_i, y_j, \theta_k)$ is partitioned into a set of overlapping segments. If $B_p(x_i, y_j, \theta_k)$ consists of u segments there will be $u(u-1)/2$ configurations (pairs of segments). Let $C_p^r(x_i, y_j, \theta_k)$ be the r^{th} configuration of boundary $B_p(x_i, y_j, \theta_k)$, where r ranges from 1 to $u(u-1)/2$.

Assume configuration C_p^r is present in an image at pose x_i, y_j, and θ_k. In general, $C_p^r(x_i, y_j, \theta_k)$ will be distorted by noise so that it will appear as some configuration $\mathbb{C}$. Let the probability that C_p^r is distorted into $\mathbb{C}$ be represented by $Pr[\mathbb{C} \mid C_p^r(x_i, y_j, \theta_k)]$. Without knowing in advance which configuration caused $\mathbb{C}$, the configuration $\mathbb{C}$ may be interpreted in several ways. Let the probability that $\mathbb{C}$ will be correctly interpreted as $C_p^r(x_i, y_j, \theta_k)$ be represented by $Pr[C_p^r(x_i, y_j, \theta_k) \mid \mathbb{C}]$. Then, the product

$$Pr[C_p^r(x_i, y_j, \theta_k) \mid \mathbb{C}] \times Pr[\mathbb{C} \mid C_p^r(x_i, y_j, \theta_k)],$$

is the probability that $C_p^r(x_i, y_j, \theta_k)$ is present in the image, appears as a configuration $\mathbb{C}$, and is correctly interpreted as configuration $C_p^r(x_i, y_j, \theta_k)$.

It is impossible to know *a priori* the form in which $C_p^r(x_i, y_j, \theta_k)$ will appear in an image due to noise distortion. It is, however, still desirable to determine the probability with which the presence of $C_p^r(x_i, y_j, \theta_k)$ will be correctly interpreted. We therefore define the saliency of configuration $C_p^r(x_i, y_j, \theta_k)$ as the probability that $C_p^r(x_i, y_j, \theta_k)$ will be present in an image and will be correctly interpreted, given that $C_p^r(x_i, y_j, \theta_k)$ could be distorted into any possible configuration $\mathbb{C}$. This definition can be written as

$$SA(C_p^r(x_i, y_j, \theta_k) \triangleq \sum_{\mathbb{C}} Pr\,|C_p^r(x_i, y_j, \theta_k) \mid \mathbb{C}| \times Pr[\mathbb{C} \mid C_p^r(x_i, y_j, \theta_k)].$$

Saliency is a measure of how unambiguously a configuration will be recognized, given all the possible distortions it can undergo due to noise. If the noise is independent of x, y, and θ, the saliency is also independent of x, y, and θ, in which case, the above definition can be rewritten as

$$SA(C_p^r) \triangleq \sum_{\mathbb{C}} Pr[C_p^r(x_i, y_j, \theta_k) \mid \mathbb{C}] \times Pr[\mathbb{C} \mid C_p^r(x_i, y_j, \theta_k)]. \tag{2}$$

The probabilities still depend on the pose x_i, y_j, and θ_k. However, it is not important which pose, only that a particular one be chosen. Varying x_i, y_j, and θ_k changes the terms that contribute in (2), but the summation remains constant because it is taken over all possible configurations $\mathbb{C}$. We will further explore this definition by individually examining the terms on the right-hand side.

The term $Pr[\mathbb{C} \mid C_p^r(x_i, y_j, \theta_k)]$ is a noise distribution for the image. It is the probability that configuration $C_p^r(x_i, y_j, \theta_k)$, when present in the image, will be distorted by noise and appear as configuration $\mathbb{C}$. The noise distribution is intrinsic to the image and not to the configuration $C_p^r(x_i, y_j, \theta_k)$.

The term $Pr[C_p^r(x_i, y_j, \theta_k) \mid \mathbb{C}]$ is the probability that

the appearance of $\mathbb{C}$ in the image will be interpreted as $C_p^r(x_i, y_j, \theta_k)$. If there are many configurations that can appear as $\mathbb{C}$ the probability will be low. If, however, only a few configurations, including $C_p^r(x_i, y_j, \theta_k)$, can appear as $\mathbb{C}$ the probability will be high. The expression for $Pr[C_p^r(x_i, y_j, \theta_k) \mid \mathbb{C}]$ can be rewritten as

$$Pr[C_p^r(x_i, y_j, \theta_k) \mid \mathbb{C}] = \frac{Pr[\mathbb{C} \mid C_p^r(x_i, y_j, \theta_k)] \times Pr[C_p^r(x_i, y_j, \theta_k)]}{Pr[\mathbb{C}]}, \quad (3)$$

where $Pr[C_p^r(x_i, y_j, \theta_k)]$ is the *a priori* probability that configuration $C_p^r(x_i, y_j, \theta_k)$ is present in the image, and $Pr[\mathbb{C}]$ is the total probability that configuration $\mathbb{C}$ appears in the image. Applying Bayes' rule to $Pr[\mathbb{C}]$ yields

$$Pr[\mathbb{C}] = \sum_{q,s,l,m,n} Pr[\mathbb{C} \mid C_q^s(x_l, y_m, \theta_n)] \times Pr[C_q^s(x_l, y_m, \theta_n)], \quad (4)$$

where $C_q^s(x_l, y_m, \theta_n)$ is the s^{th} configuration of boundary $B_q(x_l, y_m, \theta_n)$. Part q is any of the set of known parts, including p, that can appear in the image. The term $Pr[\mathbb{C} \mid C_q^s(x_l, y_m, \theta_n)]$ is the probability that given configuration $C_q^s(x_l, y_m, \theta_n)$ is present in the image, it appears as $\mathbb{C}$. The term $Pr[C_q^s(x_l, y_m, \theta_n)]$ is the *a priori* probability that $C_q^s(x_l, y_m, \theta_n)$ is present in the image. We have assumed in this expansion that the segments in a configuration come from the same part, and not from the accidental alignment of segments of two or more parts. More precisely, an *accidental alignment* occurs when a configuration of segments from two or more parts happens to fall in a relative position that resembles a configuration of segments from a single part. Strictly speaking (4) should also include terms of the form $Pr[\mathbb{C} \mid S_{q_1}^i S_{q_2}^j] \times Pr[S_{q_1}^i S_{q_2}^j]$ and higher order joint probabilities, where $S_{q_1}^i$ and $S_{q_2}^j$ are the i^{th} and j^{th} segments of different parts q_1 and q_2. Accidental alignment would cause some of these terms to be non-zero. We assume that *accidental* alignments have negligible probability. With this assumption (3) becomes

$$Pr[C_p^r(x_i, y_j, \theta_k)] \mid \mathbb{C}] = \frac{Pr[\mathbb{C} \mid C_p^r(x_i, y_j, \theta_k)] \times Pr[C_p^r(x_i, y_j, \theta_k)]}{\sum_{q,s,l,m,n} Pr[\mathbb{C} \mid C_q^s(x_l, y_m, \theta_n)] \times Pr[C_q^s(x_l, y_m, \theta_n)]}. \quad (5)$$

If we assume, for the moment, that a part is equally likely to be at any pose in the image then $Pr[C_q^s(x_l, y_m, \theta_n)]$ is a constant independent of x, y, and θ. This constant is proportional to,

$$\frac{\textit{the frequency of parts of type } q \textit{ in images}}{\textit{the number of digitized poses}}.$$

To simplify the discussion we will assume that the parts occur equiprobably; therefore, the constant terms are equal and, thus, cancel one another. This results in the following,

$$Pr[C_p^r(x_i, y_j, \theta_k) \mid \mathbb{C}] = \frac{Pr[\mathbb{C} \mid C_p^r(x_i, y_j, \theta_k)]}{\sum_{q,s,l,m,n} Pr[\mathbb{C} \mid C_q^s(x_l, y_m, \theta_n)]}. \quad (6)$$

Substituting (6) into (2), the saliency of $C_p^r(x_i, y_j, \theta_k)$ becomes

$$SA(C_p^r) = \sum_{\mathbb{C}} \frac{Pr[\mathbb{C} \mid C_p^r(x_i, y_j, \theta_k)]^2}{\sum_{q,s,l,m,n} Pr[\mathbb{C} \mid C_q^s(z_l, y_m, \theta_n)]}. \quad (7)$$

If we assume, for the moment, that no noise is associated with the boundaries of the image, equation (7) can be greatly simplified. Let $I(e \in A)$ represent an indicator function whose value is 1 when e is an element of the set A and whose value is 0 otherwise. In the noiseless case, if configuration $C_q^s(x_l, y_m, \theta_n)$ is present in an image it will appear as $C_q^s(x_l, y_m, \theta_n)$; thus, $Pr[\mathbb{C} \mid C_q^s(x_l, y_m, \theta_n)]$ becomes an indicator function $I(\mathbb{C} \in \{C_q^s(x_l, y_m, \theta_n)\})$ defined for the singleton set $\{C_q^s(x_l, y_m, \theta_n)\}$. Substituting for $Pr[\mathbb{C} \mid C_p^r(x_i, y_l, \theta_k)]$ and $Pr[\mathbb{C} \mid C_q^s(x_l, y_m, \theta_n)]$ in (7) we obtain

$$SA(C_p^r) = \sum_{\mathbb{C}} \frac{I(\mathbb{C} \in \{C_p^r(x_i, y_j, \theta_k)\})^2}{\sum_{q,s,l,m,n} I(\mathbb{C} \in \{C_q^s(x_l, y_m, \theta_n)\})} = \frac{1}{\sum_{q,s,l,m,n} I(C_p^r(x_i, y_j, \theta_k) \in \{C_q^s(x_l, y_m, \theta_n)\})}.$$

By summing over all x_l, y_m, and θ_n we are, in effect, moving part q so that each of its configurations, s, is compared with the fixed configuration $C_p^r(x_i, y_j, \theta_k)$. The resulting saliency is the inverse of the frequency with which configurations $C_q^s(x_l, y_m, \theta_n)$, that are identical to configuration $C_p^r(x_i, y_j, \theta_k)$, occur in the set of parts. This agrees with our earlier informal definition of saliency.

Now, returning to the noisy situation, we observe that if the noise distribution $Pr[\mathbb{C} \mid C_q^s(x_l, y_m, \theta_n)]$ is known, the saliency for configuration $C_p^r(x_i, y_l, \theta_k)$ can be calculated from (7). On the other hand, if the noise distribution is not known but is characterized by a fit tolerance, D, the expression on the right in (7) can be approximated as follows.

Denote by $\{\mathbb{C}\}_D$ the set of configurations with segments which are in the same relative pose as the segments of a configuration $\mathbb{C}$ and which are individually within a tolerance D in $\theta - a$ space of corresponding segments of $\mathbb{C}$. This set is simply the configurations that match $\mathbb{C}$ in the sense defined after (1). The set can also be viewed as a radius D sphere in configuration space centered on $\mathbb{C}$. Configuration space contains the set of all possible configurations. Points in this space include all the configurations C_q^s at all poses. The metric is the fit given by (1). We assume a constant density of configurations in configuration space, and let the number of configurations in a radius D sphere in configuration space be N, a constant. Then, we can approximate $Pr[\mathbb{C} \mid C_q^s(x_l, y_m, \theta_n)]$ by the term $1/N \times I(\mathbb{C} \in \{C_q^s(x_l, y_m, \theta_n)\}_D)$. In other words, we have assumed that the probability that a configuration, $\mathbb{C}$, can be distorted enough to fall outside of the radius D sphere about $C_q^s(x_l, y_m, \theta_n)$ is 0. Similarly, we approximate $Pr[\mathbb{C} \mid C_p^r(x_i, y_j, \theta_k)]$ by $1/N \times I(\mathbb{C} \in \{C_p^r(x_i, y_j, \theta_k)\}_D)$.

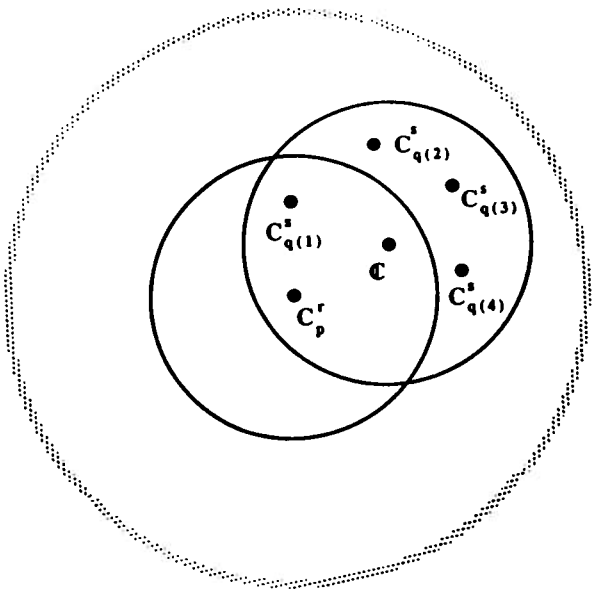

Fig. 12. Spheres in configurations space.

Substituting the above two terms into (7) we obtain,

$$SA(C^r_p) \approx \sum_{\mathbb{C}} \frac{\left(\frac{I(\mathbb{C} \in \{C^r_p(x_i, y_j, \theta_k)\}_D)}{N}\right)^2}{\sum_{q,s,l,m,n} \frac{I(\mathbb{C} \in \{C^s_q(z_l, y_m, \theta_n)\}_D)}{N}}$$

$$\approx \frac{1}{N} \times \sum_{\mathbb{C} \in \{C^r_p(x_i, y_j, \theta_k)\}_D} \frac{1}{\sum_{q,s,l,m,n} I(\mathbb{C} \in \{C^s_q(x_l, y_m, \theta_n)\}_D)}. \quad (8)$$

In the denominator of (8) configuration $\mathbb{C}$ is held fixed while the sum over $q, s, l, m,$ and n selects sets $\{C^s_q(x_l, y_m, \theta_n)\}_D$ that contain $\mathbb{C}$. Thus, the summation counts the number of spheres that contain a particular value of $\mathbb{C}$. This sum is equal to the sum

$$\sum_{q,s,l,m,n} I(C^s_q(x_l, y_m, \theta_n) \in \{\mathbb{C}\}_D). \quad (9)$$

The equality follows from the observation that the number of radius D spheres centered on the possible values of $C^s_q(x_l, y_m, \theta_n)$ that contain a particular $\mathbb{C}$ is the same as the number of centers $C^s_q(x_l, y_m, \theta_n)$ within a sphere of radius D centered on the particular value of $\mathbb{C}$. Substituting (9) into (8) yields

$$\frac{1}{N} \times \sum_{\mathbb{C} \in \{C^r_p(x_i, y_j, \theta_k)\}_D} \frac{1}{\sum_{q,s,l,m,n} I(C^s_q(x_l, y_m, \theta_n) \in \{\mathbb{C}\}_D)}. \quad (10)$$

Diagrammatically, the C^s_q terms counted in the denominator of (10) are shown in Figure 12. The solid circles represent the radius D spheres. Therefore, configurations within the circle centered on $\mathbb{C}$ are those within a D tolerance of $\mathbb{C}$.

If we assume that the density of $C^s_q(x_l, y_m, \theta_n)$ configurations is locally constant within a $2D$ sphere centered on C^r_p (the dashed circle in Figure 12), then we can obtain an estimate of the number of C^s_q's within D of $\mathbb{C}$ by the number of C^s_q within D of C^r_p. Substituting this number in (10) yields,

$$\frac{1}{N} \times \sum_{\mathbb{C} \in \{C^r_p(x_i, y_j, \theta_k)\}_D} \frac{1}{\sum_{q,s,l,m,n} I(C^s_q(x_l, y_m, \theta_n) \in \{C^r_p(x_i, y_j, \theta_k)\}_D)}.$$

Since, the sum in the denominator is now approximated by a sum independent of $\mathbb{C}$ and the cardinality of $\{C^r_p(x_i, y_j, \theta_k)\}_D$ is simply N, we can eliminate the summation over $\mathbb{C}$ to yield the following modified definition of saliency:

$$\mathbf{SA}(C^r_p) \triangleq \frac{1}{\sum_{q,s,l,m,n} I(C^s_q(z_l, y_m, \theta_n) \in \{C^r_p(x_i, y_j, \theta_k)\}_D)}. \quad (11)$$

This is the working definition that we use to calculate saliency during training. In other words, we approximate the saliency of configuration $C^r_p(x_i, y_j, \theta_k)$ as the inverse of the frequency of all configurations of the set of parts that have segments in the same relative pose as those of $C^r_p(x_i, y_j, \theta_k)$ and which are within a D tolerance of $C^r_p(x_i, y_j, \theta_k)$.

There are two points to discuss before concluding this section on saliency. First, in the special case of symmetric parts we are generally uninterested in which of the equivalent symmetric poses the part is found. For example, if a part has n rotational symmetries, n of its poses are equivalent to us. In this case the saliency of any configuration of the part should be modified by multiplying its normal saliency by the symmetry of the part. In other words, the saliency of a configuration C^r_p would become $n \times \mathbf{SA}(C^r_p)$. Clearly, this saliency is defined with respect to the part—the same configuration in the other parts in the part set, i.e. parts without n rotational symmetries, would not have the same value of saliency. Second, if there is *a priori* knowledge about the frequency of occurrence of each part in typical application images, the saliency of part-configurations can be weighted by a normalized frequency of occurrence factor. This factor can be accounted for by rederiving (11) from (5), but with the term $Pr[C^s_q(x_l, y_m, \theta_n)]$ now a function of the part q.

III.2 Training

Training is the process of determining the saliencies of the configurations of a set of parts. Assume that we wish to determine the saliencies of the configurations of boundary, B_p. An obvious approach is to start by comparing all of the configurations of B_p to those of another boundary, B_q, as was done in Figure 10. This is inefficient. If there were u_p segments in boundary B_p and u_q segments in B_q we would compare $\frac{u_p(u_p-1)}{2}$ configurations in B_p with $\frac{u_q(u_q-1)}{2}$ configurations in B_q for a total of $\frac{u_p(u_p-1)}{2} \times \frac{u_q(u_q-1)}{2}$ comparisons. Since

all parts must be compared there would be a grand total of

$$\sum_{p \in P} \frac{u_p(u_p - 1)}{2} \times \sum_{q \in P} \frac{u_q(u_q - 1)}{2}, \qquad (12)$$

where P is the set of known parts. Even though this is a one-time off-line computation, it is unacceptably inefficient.

Instead the following approach is taken. The segments of B_q are compared to the segments of B_p. If a segment of B_q at pose (x_l, y_m, θ_n) matches a segment $S_p^r(x_i, y_j, \theta_k)$ of B_p, the pose (x_l, y_m, θ_n) and the identity of S_p^r, i.e., the index r, are stored in a *match table*. The match should satisfy a D tolerance. As before, we consider the configuration to be fixed at (x_i, y_j, θ_k) while the segments of B_q are moved. The match table is implemented as a hash table with the ordered triple (x_l, y_m, θ_n) as the primary key. A key may have multiple indices, r, stored at its associated table location. After all segments of B_q have been matched to those of B_p, the match table is searched for pairs of segments of B_p which matched B_q at the same pose. These are simply pairs of indices at the same key. If two segments of B_p match two segments of B_q at the same pose (x_l, y_m, θ_k) then the configuration of the two segments of B_p must match a configuration of segments of $B_q(x_l, y_m, \theta_n)$. Thus, searching the match table for pairs of indices at the same key is equivalent to searching for matching configurations.

A 2-dimensional array, indexed by the identities of pairs of segments in B_p, is used to record the frequency with which configuration-pairs match the boundary B_q. The array elements are initially zero. Each time a configuration of B_p is found which matches a configuration of B_q, the element of the array corresponding to the pair is incremented.

After the match table has been completely searched, it is cleared and the segments of another boundary, B_q, are matched to the segments of B_p. This is repeated for all q, including p itself. Each time the array of frequencies is updated to reflect the number of matching configurations. When the segments of all parts have been matched to the segments of B_p, the reciprocals of the elements of the array yields the saliencies of the configurations of B_p. The configurations, C_p^r, with their associated saliencies, $\mathbf{SA}(C_p^r)$, form the part-model of p. In most applications only those configurations, C_p^r, with $\mathbf{SA}(C_p^r) \approx 1$ are retained for the salient features. The whole procedure is repeated for each part. It is straightforward to show that the number of comparisons required by the training procedure is given by

$$\sum_{p \in P} \sum_{q \in Q} u_p u_q.$$

This compares favorably with (12).

IV. USING SALIENT RECOGNITION FEATURES TO SOLVE THE POP PROBLEM

If a particular part is sought, an efficient strategy searches for configurations in order of decreasing saliency. As an example, consider searching for the rectangle shown in Figure 9. Model configurations which include the notch as a segment have the largest values of saliency and should be compared first to the POP image. If no such pair can be found, a model configuration with less saliency, e.g., corner A and corner C, should then be compared to the image. This process is continued until the part is found or no untried model configurations are left. If the latter situation occurs we assume the part is not present or is totally hidden. If more than one part is sought, for example a subset of the set of parts, an efficient strategy is to search for all the model configurations from all of the parts in the subset in order of decreasing saliency.

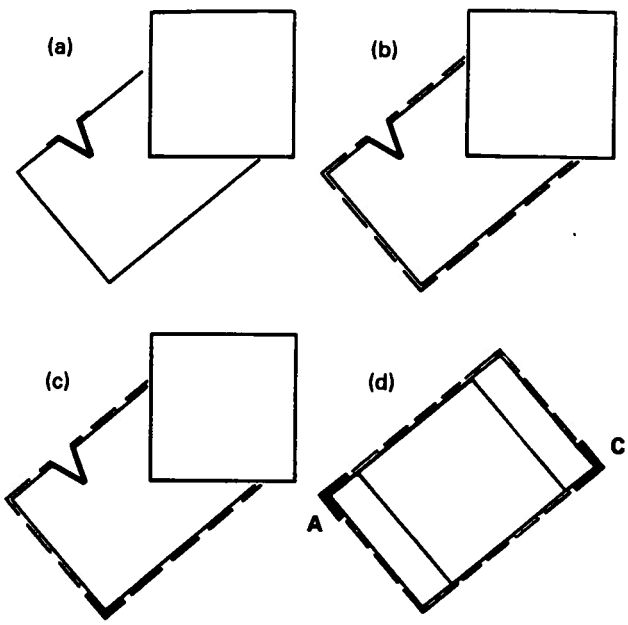

Fig. 13. Locating a part.

Searching for matches to model configurations of a particular saliency can best be done by first searching for the model segment that occurs most often in the configurations. The search for individual segments can be done in the manner outlined in the previous section, and illustrated by the following.

If the segment has a curvature extremum, we align the curvature extremum of the segment with an extremum in the image boundary before comparing the segments. On the other hand, if the segment has no extremum, it must be compared to all segments of the image boundary. In both cases, comparison is performed in the $\theta - a$ representation. If a good match is found between a model segment and an image segment (see the notch Figure 13(a)), the rotation and translation necessary to align the two segments is computed by performing a least squares fit of the two segments in cartesian space. The rotation and translation are applied to the entire boundary of the part (see dotted outline in Figure 13(b)) and the transformed boundary is used as a guide in searching for the second segment of a configuration with high saliency (see the lower corner in Figure 13(c)). The saliency provides an estimate of the probability that the correct part at the correct pose has been located. In our example, if the notch in the rectangle were not visible in

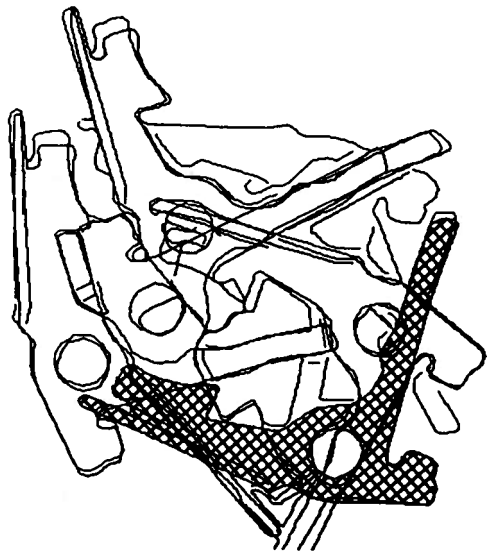

Fig. 14. Finding a part in a bin of parts.

the POP image (see Fig. 13(d)), the algorithm locates the rectangle using a less salient pair (corner A and corner C in Fig. 13(d)) and reports that the probability of it having found the correct pose is $\frac{1}{2}$, i.e., the saliency of A and C.

As a second example the technique was applied to the bin of identical overlapping parts shown in Figure 1. Using the new recognition method it was possible to count, with a high degree of accuracy, the number of parts in the scene. All of the seven parts present were correctly located when the method was tried. Figure 14 shows the location of one of them.

Although space does not permit detailed evaluation of the new technique two points are worth noting. First, the time to recognize a set of parts from a bin containing about a dozen parts is less than 1 second, if a VAX 11/780 minicomputer is used. This compares favorably with other common approaches. Second, experiments were run against other common approaches to compare recognition ability. In all cases the method presented here identified all the parts in the bins that it inspected. This robustness was unmatched by any of the others.

A variation on the above search strategy, due to Knoll[19], is to maintain pointers back to model boundaries of all the parts that contained each model configuration. The search starts with model configurations (the configurations used in ref. 19 consist of single segments) that have non-maximal saliencies. When such configurations are located in the image, their possible interpretations are determined by fitting to the image all the boundaries from which the model configurations may have come, and then selecting the boundary and pose with the best fit as the correct interpretation. This works best if there are a large number of parts in the set and only a few are expected to appear in any image; and if fitting the entire boundary can be done efficiently.

V. CONCLUSION

This paper has focused on the concept of saliency, a new concept that allows improved recognition ability, especially in the presence of occlusion. A method was presented for automatically generating salient features, or configurations, from primitive features. A strategy was presented for using salient configurations to solve the POP recognition problem. Space has not permitted us to present a complete evaluation of the method. Details are available in ref. 2 where the performance of the method has been evaluated for bins containing a mixture of parts as well as identical parts.

References

1. J. Mattill, "The Bin of Parts Problem and the Icd-Box Puzzle" *Technology Review* **78,** No. 7, 18–19 (June 1976).
2. J.L. Turney, "Recognition of Partially Occluded Parts", *Ph.D. Thesis*, University of Michigan (1986).
3. P.M. Merlin and D.J. Farber, "A Parallel Mechanism for Detecting Curves in Pictures" *IEEE Trans. on Computers* **C-24,** No. 1, 96–98 (Jan. 1975).
4. D.H.Ballard, "Generalizing the Hough Transform to Detect Arbitrary Shapes" *Pattern Recognition* **13,** No. 2, 111–122 (1981).
5. W.S. Rutkowski, "Recognition of Occluded Shapes Using Relaxation" *Computer Graphics and Image Processing* **19,** 111–128 (1982).
6. R.B. Kelly, H.A.S. Martins, J.R. Birk and J.D. Dessimoz, "Three Vision Algorithms for Acquiring Workpieces from Bins" *Proc. IEEE* **71,** 803–820 (July, 1983).
7. L. Jacobson and H. Wechsler, "Invariant Image Representation: A Path Toward Solving the Bin-Picking Problem" *Proc. IEEE Int'l Conf. on Robotics* 190–199, (March, 1984).
8. H. Blum and R.N. Nagel, "Shape Description Using Weighted Symmmetric Axis Features" *Pattern Recognition* **8,** No. 3, 167–180 (1978).
9. M. Brady and H. Asada, "Smoothed Local Symmetries and Their Implementation" *Intern. J. Robotics Res.* **3,** No. 3, 36–61 (1984).
10. R.C. Bolles and R.A. Cain, "Recognizing and Locating Partially Visible Objects: The Local-Feature-Focus Method", *Robot Vision,* ed. A. Pugh (Springer Verlag, New York, 1983).
11. S. Berman, P. Parik and C-S. G. Lee, "Computer Recognition of Overlapping Parts Using a Single Camera" *Proc. IEEE Conf. on Pattern Recognition and Image Processing* 498–503 (June, 1982).
12. H. Tropf, "Analysis-by-Synthesis Search for Semantic Segmentation – Applied to Workpiece Recognition" *Proc. 5th Int'l Conf. on Pattern Recognition,* Miami Beach 241–244 (Dec., 1980).
13. M.W. Koch and R.L. Kashyap, "A Vision System to Identify Occluded Industrial Parts" *Proc. IEEE Conf. on Pattern Recognition and Image Processing* 55–60 (1985).
14. G. Stockman, S. Kopstein and S. Benett, "Matching Images to Models for Registration and Object Detection via Clustering" *IEEE Trans. on Pattern Analysis and Machine Intelligence* **PAMI-4,** No. 3, 229–241 (May, 1982).
15. A.N. Ayache and O.D. Faugeras, "A New Method for Recognition and Position of 2-D Objects" *Proc. 7th Int'l Conf. on Pattern Recognition* **2,** 1274–1277 (Aug., 1984).
16. M.M. Lipschutz, *Schaum's Outline of Theory and Problems of Differential Geometry,* (McGraw-Hill, New York, 1969).

17. J.L. Turney, T.N. Mudge and R.A. Volz, "Recognizing Partially Occluded Parts" *IEEE Trans. on Pattern Analysis and Machine Intelligence* **PAMI-7,** No. 4, 410–421 (July, 1985).
18. J.L. Turney, T.N. Mudge and R.A. Volz, "Recognizing Partially Hidden Objects" *Proc. IEEE Int'l Conf. on Robotics and Automation* 48–54 (March, 1985).
19. T.F. Knoll and R. Jain, "Recognizing partially visible objects using feature indexed hypothesis" *Center for Research on Integrated Manufacturing Report* No. RSD-TR-10.85 (July, 1985).