The Ensemble Representation

Yerucham Shapira
Dept. of Electrical and Computer Engineering
Ecole Polytechnique of Montreal
P.O.Box 6079, Station A
Montreal, Que., Canada H3C 3A7,
E-mail: shapira@ai.polymtl.ca

Abstract

The ensemble representation is presented as an alternative approach to the problems of image segmentation and of inferring the structural organization of images. According to this approach the consistency requirement, usually imposed on the results of image segmentation, is relaxed, and the elements of the representation are extracted individually. This gives rise to a set of structural elments only a part of which represent actual entities in the viewed scene. The selection of a subset of the elements of the representation to form a coherent interpretation of the image is deferred to a later stage in which additional information, such as, consistency of a number of cues or higher level knowledge concerning the scene, is integrated. The primary type of elements for the ensemble representation are closures, *i.e.*, contour segments delineating image regions. The importance of this type of elements is supported by psychological evidence and has also been demonstrated in the context of object recognition. An iterative method for extracting closures from edge-maps is presented, and a parallel architecture implementing it is discussed.

1 Introduction

The task of delineating specific regions in the image is usually denoted as image segmentation or part decomposition, and a number of methods have been proposed for accomplishing it. These methods generally fall into two categories depending on the application in mind. In the context of bottom-up scene interpretation, the process of separating the image into distinct regions is usually interwined with that of inferring the particular attributes for those regions, that is, regions are delineated according to differences in the sought attributes. Since images are usually noisy, a smoothings operation is involved with computing the properties of given areas in the image, and the boundaries of the smoothing process and the resulting properties are therefore determined simultaneously. This usually involves the minimization of some cost function measuring the consistency of the smoothed property with the raw data, and the smoothness of that property. This minimization task is often a difficult one.

The second context is that of object recognition. Here the image is usually segmented using certain properties, assumed to be shared by many objects. Hoffmann and Richards [1984], for instance, rely on the concavity points that are usually observed at the joints of different parts. Another property used for delineating objects and object parts is symmetry [Blum 1973]. Connell and Brady [1987] have demonstrated how object parts inferred according to their symmetry can be used for object recognition. One problem with this type of approach is that universal properties, distinguishing "good" parts in images, are difficult to find. Therefore, the above methods are not suitable for all types of 6bjects and often yield ambiguous results and over-segmentation of images.

Decomposing images of complex scenes into structurally meaningful regions (*e.g.*, objects and parts) and generating a representation in which these elements are made explicit is closely related to the issue of perceptual organization (see, for instance, [Witkin and Tenenbaum 1986]). The importance of this intermediate level of visual processing is well established, and recent recognition approaches that rely on two dimensional input (*e.g.*, [Fischler and Bolles 1981; Lowe 1985; Shapira and Ullman 1991]), emphasize such a level even further as being sufficient for many aspects of object recognition.

The approach used by the existing methods for image decomposition can be denoted as "coherent", namely, searching for a single, self-consistent structure in the image. The consistency requirement can be roughly interpreted as imposing that each image primitive (*e.g.*, each line) be incorporated in a structural element, and that the structural elements would not overlap. This coherent approach usually yields re-

sults that are highly sensitive to changes in the input and cannot contribute to a reliable interpretation of the image.

The ensemble approach

It appears that the difficulties encountered devising a reliable scheme for image segmentation, both in bottom-up and knowledge-based contexts, originate from the same reason — committing on a definite structural organization of the image too early in the course of processing.

A complementary approach, stressing the least commitment principle [Marr 1982], is proposed here. It assumes that bottom-up grouping processes do not suffice to generate of a coherent interpretation of the image. This task is therefore divided into two stages. First, a representation denoted as *the ensemble representation* is constructed. This representation consists of many elements of global nature that are extracted from the image (*e.g.*, blobs, distinguished regions, etc.). The elements would potentially, but not necessarily, be included in the eventual description of the image structure. It should be emphasized that the elements of the ensemble representation are extracted individually, thus relaxing the consistency requirement which is difficult to satisfy. In fact, the elements may well be contradicting, or rival, in the sense that different elements may share some underlying lower level primitives (*e.g.*, edges) or be partially overlapping. In this sense the ensemble representation itself may be regarded as incoherent, indicating a (large) number of different possible "organizations", or interpretations, of the image.

Some typical elements of the ensemble representation found for the edge map of a scene that contained a book and a teddy-bear are shown in Fig. 4. Note that some of the elements do not correspond to meaningful entities in the scene (4a, 4b). Fig. 4c, on the other hand, does correspond to such an entity — one of the book's sides. The element in 4(e) illustrates possible incoherencies of the representation, it consists of some of the edges of the the element in 4c as well as combining two objects.

In the second stage, task specific processes are applied to the ensemble representation for selecting one particular and unique interpretation of the image (*i.e.*, a particular subset of the representation elements) among the many possible. This selection process can rely on two types of information. The more significant elements may be selected by testing for homogeneity and smoothness of intrinsic image properties such as texture, color, depth, and orientation across them. The closures can also be selected using higher level considerations such as "minimality of shape" [Attneave 1954], or prior knowledge such as the quality of match with components of internally stored object models (in the course of recognition).

It should be noted that the interesting phenomena of "rival organizations" that can be perceived for some stimuli, and of interpretation that are strongly affected by higher level knowledge (*e.g.*, the spotted dog image by R. James [Marr 1982, p. 101]), may be well explained in the context of the ensemble approach.

The reminder of the paper is organized as follows. In the following section we introduce the principal type of elements of the ensemble representation, denoted as closures, along with listing several other possible types of elements. In Section 3 we present an algorithm for extracting closures from edge-maps of images and discuss a parallel architecture for implementing it. A short summary concludes this paper.

2 The Primary Elements – Edge-Based Regions

The perceived property of closure

So far we have not specified what elements would be appropriate for the ensemble representation. Different types of constructs can qualify for this purpose, and, in fact, more than one type will be required for realistic scenes. In the following we restrict ourselves to the edge-map domain. That is, the elements of the ensemble representation discussed bellow are all defined by the edges extracted from the image. The most natural and useful elements for the representation in this domain would be image regions of relatively "simple" shape, delineated by edges in the image. The fact that visual data is often noisy and incomplete, suggests that the contour segments grouped together to delineate regions in the image need not necessarily be exactly closed. Indeed, even contours whose shape is as open as the letter 'C' are perceived to define an enclosed area. We denote this type of contours as *closures*.

In a recent set of experiments, Elder and Zucker [1991] have examined the effects of the property of closure on shape perception, by using it as a control parameter in a target/distractors discrimination paradigm. The target and the distractors in these experiments were contours of different shapes, but of the same degree of closure. The role of the closure property was evaluated by examining the discrimination performance as the degree of closure of the target and distractor contours varied simultaneously.

Their findings indicate two conclusions regarding the role of this property in shape perception. The primary conclusion is that the ability to discriminate between contour shapes depends in a smooth and continuous way on the extent by which they are "closed". The more closed the contours, the faster it is to discriminate their shape. In fact, their results revealed that if the contours were closed enough, the speed by which a target of a given shape was located among distractors of a different shape, depended only weakly

on the number of these distractors, and was within the range associated with preattentive perception. When the contours were not sufficiently closed, the performance depended much stronger on the number of distractors, and was in the range of attentive search[1].

The second conclusion was that inferring the closure property for contours involves a low level type of computations. This has been indicated by the fact that when the contours included segments of reversed contrast, these segments did not contribute to increase the speed of shape discrimination, implying that they may have not contributed to the closure of the contours. As processes involving more global grouping mechanism (also denoted as "long range") are thought to be invariant under contrast reversal, this may indicate that inferring closure relies on low level ("short range") computations.

The important consequence regarding the ensemble representation is that contours that delineate well-defined image regions (*i.e.*, sufficiently closed contours) are important for shape perception whether they are perfectly closed or not.

Closures as the elements of the ensemble representation

In the following section we describe an algorithm for extracting *closures* from edge maps of images. The closures, as defined here, are contours that delineate connected and generally convex image regions. The connectness property of the image regions implies that a physical identity whose image is non-connected would be considered as composed of more than one element. Consider for example a puddle that has partly dried out and, as a result, has been separated into two different sections. A more common example is a partly occluded object or object part. In some cases, the entire object may still yield a large closure encompassing its different sections (due to the smoothness of its overall silhouette contour), but this requirement is not imposed on the representation elements. The requirement that the regions delineated by the closures be "generally convex" is motivated by the observation made by Hoffman and Richards [1984] that joints between different parts of objects are usually points of deep concavity in the silhouette of the object, and that the visual system tends to use such locations for segmenting non-familiar objects.

An object recognition application

The relevance of closures to higher level tasks, was examined in the context of object recognition. A pictorial approach for classifying objects according to similarity between their shape and that of a coarse class model has been devised in [Shapira and Ullman 1991]. Correct classification of objects was obtained there by aligning internal models of classes of objects with the image and, then, deforming the parts defined (only) in these internal models according to corresponding features matched in the model and the image. A preliminary algorithm for extracting structural elements in images was examined in [Shapira 90], yielding few hundreds of image regions as possible building blocks for the ensemble representation of the images. The model parts were then matched to the regions found by the preliminary algorithm yielding promising results. Establishing correspondence between model parts and global elements in the image facilitated detailed analysis of the shape similarity between the image and the internal model, thus, enabling object classification. Partial results of matching model parts with the elements of the ensemble representation extracted from the image are shown in Figure 1.

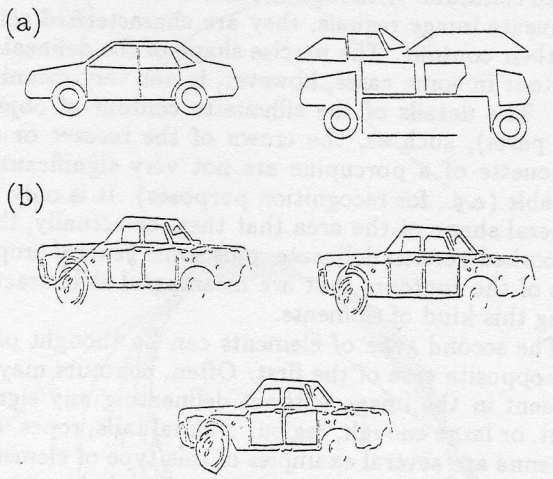

Figure 1: Using the ensemble representation for object classification. (a) The model of the "car" class of objects and the parts defined in it. (b) The closures found in an image of a particular car toy, and matched to the model parts. The front part and the windows (left), the top part and the main chunk (right), and the doors and the front wheel (bottom). The back part and the back wheel of the model were not successfully matched (from [Shapira 1990]).

Other types of elements

Closures are not the only possible elements for the ensemble representation. In many cases they would not suffice to convey the entire structural organization underlying the image. Before proceeding to a description of a method for extracting the closures we list three other types of possible elements. Part of the elements listed below were denoted by Ullman [1989] as "abstract descriptors".

The first type consists of regions with highly frac-

[1] A continuous closure property, rather than a binary one, may suggest why clear evidence for the ability to discriminate preattentively between closed and open closure has not been found.

tured contours. Although the closures described above delineate image regions, they are characterized solely by their contour. The precise shape of the delineating contour in some cases, however, is not very meaningful. The details of the silhouette contour of objects (or parts), such as, the crown of the rooster or the silhouette of a porcupine are not very significant or reliable (*e.g.*, for recognition purposes). It is only the general shape of the area that they, or actually, their smoothed version delineate, plus some *general* properties of the contours that are meaningful as characterizing this kind of elements.

The second type of elements can be thought of as the opposite case of the first. Often, contours may be present in the image without delineating any significant, or large enough, region. Animal tails, ropes, and antenna are several examples of this type of elements.

The third type consists of region based elements. A region of the image may be distinguished from their surroundings by some particular characteristic type of edges (or contour segments) present in it without any contour surrounding that region. Textured regions, such as, the radiator panel of (old) cars is one example of such regions.

3 The Closure Extraction Algorithm

Closures have been defined in the previous section as contours having a simple shape and delineating definite image regions. Here we present an algorithm for extracting closures from edge maps. The algorithm proceeds by repetitively merging pairs of contour segments, each represented by a single elliptic arc, and approximating them by extended contour segments. Elliptic arcs seem a natural choice for this purpose since they have closed, smooth, and convex shape and allow for variable location, size, elongation, and orientation of the closure they represent. Hence, a good approximation of contour segments by elliptic arcs implies that those contour segments are good candidates for being closures. The pairs of contour segments to be merged are determined by associating *influence zones* with each of the contour segments at any given moment. These zones consist of the pixels tracing the contour, and a curved cone-shaped area at each end-point of the contour. Pairs of contour segments whose influence zones have overlapping sections are merged together. Each merger is then evaluated for deciding whether to accept the result as representing the two original segments, as well as, whether the resulting contour is significant and closed enough for being considered a closure. The algorithm is described in more details below, and a parallel implementation is discussed.

Identifying the contour segments to be merged

The input for the extraction procedure was a list of (parametric) curves obtained from the edge map of the image. This parametric representation of the edge map was obtained by a procedure for tracing sequences of edge pixels and then segmenting and approximating them by straight lines and circular arcs [Shapira 1992]. The iterative part starts by associating an *influence zone* with each of the curve segments in this list. In subsequent iteration cycles, the influence zones are computed for the elliptic arcs representing the contour segments that have been formed in the process. Each influence zone consists of three parts: the pixels that define the contour segment itself, and two cone-like regions extending from each end-point of the segment. This cone-like regions are obtained by extending the curve representing the contour segment beyond its end-point and rotating the extended portion around that end-point, scanning a fixed angle. This procedure yields a section of a circle when the curve is a straight line and a similar shape, only having a curved axis, when the curve is a circular or an elliptic arc. The contour segments were extended by a 1/3 of their length to each direction, and the angle scanned was $\pm 10°$ around their end-point orientations. The influence zone obtained for an elliptic arc is shown in Figure 2.

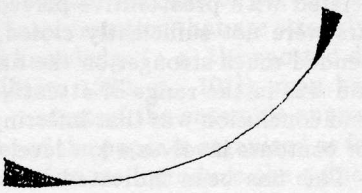

Figure 2: The influence zones for an elliptic arc. The zone consists of three parts: the actual contour and a curved "cone-like" region at each end-point (the end point regions would have had a straight axis had the curve been a straight lines).

The influence zones are registered in a *board* consisting of *cells*, each of which corresponding to a pixel of the original image. A zone is registered in each of the cells it passes through by two integers: an index pointing to the particular contour segment with which it is associated, and a second number, identifying which of the zone components actually passes through the cell: the contour itself, or one of the end point regions. When zones of more than one contour segment coincide at a given board cell, the list of pointers to those curve segments will be registered in that board cell (along with a list of numbers identifying the zone components). Each pair of contour segments pointed at is marked for a future merging attempt to be made.

The actual architecture is a little different than described above due to memory size considerations.

Since the same set of influence zones is often found in neighboring board cells, a table of the *prototypes* of board cells is constructed in which each prototypical board cell (*i.e.*, that contains a given set of influence zones registered in it) is represented by a single entry. Thus each board cell contains only a single pointer to a particular entry in the prototype table, and only a smaller number of entities (the entries in the prototype table) have the more complex structure, namely, the two lists of pointers and identifiers for the zone components.

Merging pairs of contours

After completing the enumeration of the pairs of contour segments whose influence zones partly overlap, an attempt is made to merge each of these pairs. This is done by sampling a number of points on each of the two curves representing the segments (20 points in total, having the same spacing on both curves) and then fitting the best elliptic arc to the set of sampled points. Ensuring that the result is an ellipse and not a different conic section (hyperbola or parabola) is not trivial, and a number of rather elaborate procedures have been devised for this purpose (see for example [Rosin and West 1990] and the references therein). Since in our case the approximation is not required to be highly accurate we employed a rather simple heuristic for obtaining the elliptic approximation. Instead of fitting a general ellipse to the set of sample points, a number of ellipses in a given set of orientations were fitted. Other conics may still appear even when the orientation of the fitted curve is determined. Using a number of orientations for which the approximation is attempted, however, increases the probability of obtaining at least a few elliptic solutions. Indeed, we had a number of elliptic results in almost all merging attempts. The ellipses obtained for different orientations are then compared (in terms of the root mean square distance of the sample points from the resulting curve) and the best curve is selected as approximating the merged contour segments. This heuristics does not yield the best fitted ellipse, but in most cases it provides a sufficiently good approximation for our purpose. An example of computing the elliptic arc approximation for merging a straight line and an elliptic arc is shown in Figure 3. Note that fitting a general ellipse failed in this case (yielding an hyperbola).

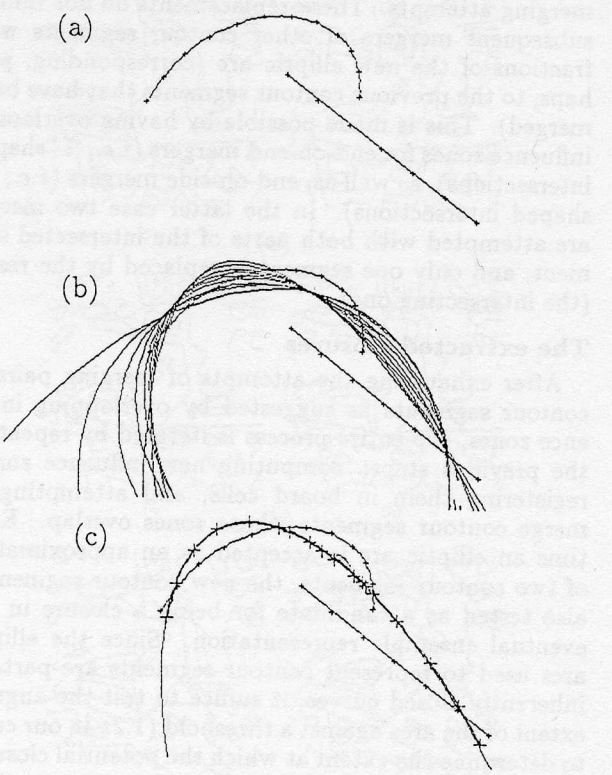

Figure 3: Approximating two contour segments by a single elliptic arc. (a) The two contour segments (a straight line and an elliptic arc) with the sample points marked on them. (b) Since fitting a general ellipse failed, a number of ellipses were fitted for a set of pre-defined orientations. For nine different orientations (out of 20 attempted) the fitted curve turned out to be an ellipse. (c) The best ellipse of those nine with the projections of the sample points marked on it.

Evaluating the merger

To evaluate the plausibility of merging two contour segments and approximating them by a single elliptic arc, a cost function is associated with the new curve and tested against a pre-set threshold. The cost function is defined as:

$$C = (1 + c_1)(1 + c_2)\sqrt{\left(w\frac{d_{rms}}{D}\right)^2 + \left(\frac{L}{l_1 \cup l_2}\right)^2}$$

where c_i is the cost associated with generation of the i-th merged contour segment (– zero for the initial curves), d_{rms} is the root mean square deviation of the sample points from their projection on the approximating ellipse, D is a combined measure of the length and overall curvature radius of the approximating ellipse (thus scaling d_{rms} by the extent by which the approximating ellipse "curves" to reach maximal proximity to the points), L is the length of the approximating elliptic arc, l_i is the length of the projection of the i-th merged curve on the approximating one, and w is a constant weight for the deviation term (set to 2.0 in the implementation). The threshold that the cost values are tested against is set to 1.0.

When a given merger of two contour segments is accepted (*i.e.*, the cost associated with it is low enough), the new elliptic arc usually replaces the two original segments and is used instead of them in subsequent

merging attempts. These replacements do not inhibit subsequent mergers of other contour segments with fractions of the new elliptic arc (corresponding, perhaps, to the previous contour segments that have been merged). This is made possible by having overlaps of influence zones for end-on-end mergers (*i.e.*, 'L' shaped intersections), as well as, end-on-side mergers (*i.e.*, 'T' shaped intersections). In the latter case two merges are attempted with both parts of the intersected segment, and only one segment is replaced by the result (the intersecting one).

The extracted closures

After exhausting the attempts of merging pairs of contour segments as suggested by overlapping influence zones, the entire process is iterated by repeating the previous steps: computing new influence zones, registering them in board cells, and attempting to merge contour segments whose zones overlap. Each time an elliptic arc is accepted as an approximation of two contour segments, the new contour segment is also tested as a candidate for being a closure in the eventual ensemble representation. Since the elliptic arcs used to represent contour segments are parts of inherently closed curves, it suffice to test the angular extent of the arcs against a threshold (1.2π in our case) to determine the extent at which the potential closures delineate image regions.

Typical closures obtained by this algorithm are shown in Figure 4. In the four parts of the figure (a–e), (i) shows the original edge segments (within the initial parametric curve approximation) that were traced back from the emerging closures, and (ii) shows the approximating contour. A typical closure having a "well-closed" shape without corresponding to any particularly important region in the image is shown in part (a), and another closure having a more peculiar shape is seen in (b). A closure that corresponds to a significant part of the object in the image is shown in part (c), and the general spot cast by shadow is captured by the closure shown in (d). The closure shown in Part (e) demonstrates the inconsistency allowed for elements of the ensemble representation — one of the book's sides is grouped together with part of the silhouette of the teddy-bear near-by. This inconsistency is discussed in Section 1.

A parallel architecture

A parallel version of the above algorithm can also be devised. Consider an architecture in which processing units are allocated to each contour segment at any time. In addition, consider another set of processing units associated with the prototypical cells in which the influence zones are registered. The number of (active) units in both sets of processors is not fixed, and they should be capable of more complex computations than those units in lower levels of visual information processing. This might have been expected for an architecture producing a considerably higher level representation than, for instance, orientation selection.

Given these two sets of processing units the algorithm can proceed by having each of the units associated with the contour segments maintain active connections to influence zones. The units associated with the prototypical cells, also connected to the influence zones, would then trigger merging attempts of the contours. In the case of successful mergers and subsequent replacement of the original contour segments, the respective contour and cell units become passive and other units are allocated to the new contours and zones. One difference between this and the implemented algorithm described above lies in the type of control of the computation of new influence zones. In our algorithm new influence zones are generated periodically as soon as *all* the merging attempts implied by the prototypical cells have been performed. This requires for a synchronization of the parallel architecture discussed here. The effects of using an asynchronic paradigm are being currently investigated.

4 summary

The task of providing image segmentation and inferring the structural organization underlying images is known to be a difficult one. The ensemble representation, proposed in this paper, presents a new approach to the problem and complies better with the least commitment principle formulated by Marr [1982]. It consists of many structural elements of several types, each extracted individually and, possibly, being inconsistent with other elements of the representation. On its own such a representation does not offer an interpretation of the image. It is only in a later stage that task-specific processes act to select a small subset of the elements of the ensemble representation, as a coherent interpretation of the image.

Closures, defined as contour segments that delineate well defined regions of the image, are argued to be the most natural and useful type of elements for the ensemble representation (within the domain of edge-maps). Psychological evidence support the importance of such contours for shape discrimination. An iterative method for extracting closures from edge-maps is presented, together with some results. The algorithm proceeds by repetitively merging pairs of contour segments and approximating them by single elliptic arcs. A parallel architecture implementing the above algorithm is discussed.

Acknowledgement

I wish to thank Paul Cohen, Greg Dudek, and James Elder for helpful discussions. I also thank Paul Rosin for providing part of the code used for preprocessing. Support for this work has been provided by the

Institute for Robotics and Intelligent Systems (IRIS) through the Network of Centers of Excellence.

References

[Attneave 1954] F. Attneave. Some informational aspects of visual perception. *Psychological Review*, 61: 183–193, 1954.

[Blum 1973] H. Blum, Biological shape and visual science, *J. Theor. Biol.*, 38: 205–287, 1973.

[Connell and Brady 1987] J.H. Connell and M. Brady, Generating and generalizing models of visual objects, *Artificial Intelligence*, 31: 159–183, 1987.

[Elder and Zucker 1991] J. Elder and S. Zucker, Contour closure and the perception of shape, Technical Report CIM-91-08, McGill Research Centre for Intelligent Machines, 1991.

[Fischler and Bolles 1981] M.A. Fischler and R.C. Bolles, Random sample consensus: A paradigm for model fitting with application to image analysis and automated cartography, *Comm. of the ACM*, 24: 381–395, 1981.

[Hoffman and Richards 1984] D. D. Hoffman and W. A. Richards. Parts of recognition. *Cognition*, 18: 65–96, 1984.

[Lowe 1985] D. G. Lowe. Perceptual organization and object recognition. *Boston: Klwer Academic Publishers*, 1985.

[Marr 1982] D. Marr, Vision, a computational investigation into the human representation and processing of visual information, *W. H. Freeman and Co., San Francisco*, 1982.

[Rosin and West 1990] P.L. Rosin and G.A.W. West, Segmenting curves into elliptic arcs and straight lines, *Proc. of the Third Intern. Conf. on Computer Vision, Japan*, 75–78, 1990.

[Shapira 1990] Y. Shapira. A pictorial approach to object classification and recognition across shape changes. *Weizmann Inst. of Science, Dept. of App. Math.*, Ph.D. Thesis, 1990.

[Shapira 1992] Y. Shapira. Approximating contours by straight lines and circular arcs, *Ecole Polytechnique de Montreal*, Technical Report, 1992.

[Shapira and Ullman 1991] Y. Shapira and S. Ullman, A pictorial approach to object classification, Proc. of the 12th Intern. Joint Conf. on Artificial Intelligence, 1257–1263, 1991.

[Ullman 1989] S. Ullman, Aligning pictorial descriptions: An approach to object recognition. *Cognition*, 32(3): 193–254, 1989.

[Witkin and Tenenbaum 1986] , A. Witkin and M. Tenenbaum, On perceptual organization, *In*: From Pixels to Predicates, Pentland A.P. (ed.), *Ablex Publishing Corp., Norwood, NJ*, 1986.

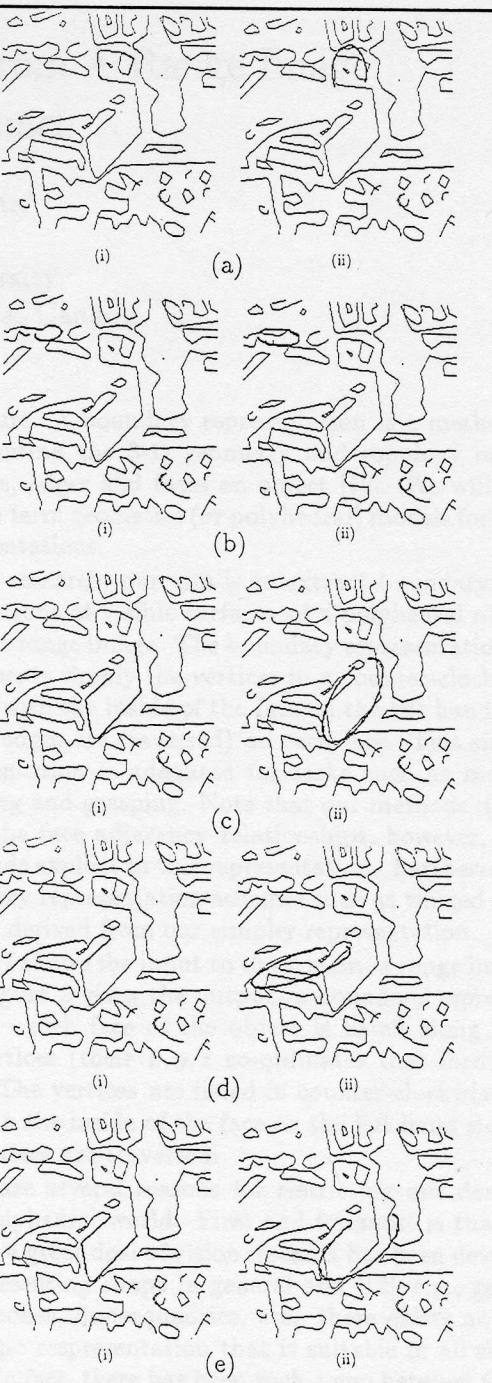

Figure 4: Typical closures extracted by the algorithm. In all part (i) is the edge lines forming its contour (in bold lines), and (ii) is the elliptic arc approximation of its contour. (a) A that do not correspond to a physically meaningful entity. (b) A closure of a less regular shape. (c) A closure corresponding to a meaningful part of the book. (d) A closure corresponding to a spot cast by shadow on the book. (e) An example of a closure that is inconsistent with the other closures that represent meaningful physical entities, such as, the one in (c).