Part segmentation of objects in real images

Guillaume-Alexandre Bilodeau, Robert Bergevin*

Computer Vision and Systems Laboratory, Pavillon Adrien-Pouliot, Université Laval,

Ste-Foy (Québec), G1K 7P4, Canada

Submitted: June 15, 2001 for Pattern recognition

* Corresponding author. Tel: *+1 418 656 2131 x5173* fax: *+1 418 656 3594*
E-mail address: bergevin@gel.ulaval.ca (R.Bergevin)

Abstract

This paper presents one significant aspect of a project aimed at the design of an image database query engine, where the images are searched at the 3D object-level. This approach is a novelty since image databases are usually searched by comparing colours, textures and 2D shapes of regions in the images. The main aspect and contribution of this paper is a new method for object part segmentation, from constant curvature contour primitives, that is suited to process images of objects in real scenes. It groups constant curvature primitives using intermediate-level geometric relationships and object outline information. A detailed description of the method is presented along with validating experiments.

Keywords: Part segmentation; Perceptual grouping; Image query; Volumetric primitives; Geons;

1. Introduction

The comparison of objects in 2D images using efficient and reliable algorithms is a problem that remains unresolved in computer vision. A similar problem is the identification of an object in an image. A value of similarity is obtained as the results of the comparison of two objects, whereas one or more identifiers are produced when an object is identified. The images are processed with common algorithms, but the results obtained are interpreted differently. The resolution of these two problems is of high interest as it permits the development of autonomous robots and efficient image database query engines.

Our work aims precisely at developing robust algorithms to model and compare 3D objects in 2D images in the context of an image database query engine. In this paper, one specific aspect is presented. It is the segmentation of a 3D object into 2D parts. In the context of this work, 2D parts are defined as regions delimited by groups of circular arcs and straight line segments (constant curvature primitives), which can later on be interpreted as the projections in the plan of simple volumetric primitives, likes cylinders and prisms.

The main contributions of this paper are the detection of parts by grouping two constant curvature primitives (CCPs) and the uses of the outline to reduce the number of possible groups. We will show that our approach is suited to process 3D objects in 2D images of real scenes by several validating experiments.

The paper is structured as followed. Section 2 gives an overview of the application. Section 3 provides a review of the literature and our basic strategy to part segmentation. Section 4 and 5 present in detail our method. Section 6 gives an analysis of results obtained with our method. Section 7 concludes the paper.

2. Overview of the envisioned application

The application under development aims at querying an image database of manufactured objects, which are arrangements of simple volumetric primitives. Images in the database are from real scenes with an object in the foreground. The only other constraint is that the object must be detectable in the image (see [1]).

Figure 1

Figure 1 shows an overview of the image database query engine under development. The shaded region represents the four algorithmic steps required to add an image or query the database. The database is composed of various 2D images of 3D objects, and their associated models. To query or add an image to the database, the user gives as input an example 2D image or a sketch of the

3D object. The image is first processed to obtain its constant curvature primitives (CCPs) map and the outline of the object (Object detection). This CCPs map is then processed to obtain parts using the outline (Part segmentation). These parts are labelled based on the possible volumetric primitives that may project onto them (Object interpretation). Parts are interpreted by volumetric primitives since the aspect of a projected 3D object may change significantly for different viewpoints (see Figure 11). At the object interpretation step, the spatial relationships between parts are also computed. Finally, the constructed model is compared with the models in the database (Model matching). If similar models are in the database, the corresponding 2D images are shown to the user. If not, the newly built model and its corresponding image may be added to the database.

Therefore, the general goal of the image database query engine is to show to the user the images in the database which resemble the most the query image, and at the same time to limit the number of false positives. The obtained images will be classified from the most to the least similar image based on the score obtained during the matching step. The topic of this paper is step 2, part segmentation. The following section reviews the literature and describes our approach to this problem.

3. Possible approaches to part segmentation

Five types of approaches have been proposed to solve the part segmentation problem. They are the approaches based on: i) symmetry axes of the parts of the object, ii) convexity points on the boundary of the object, iii) minimum description length criterion, iv) primitives cycles, and v) perceptual grouping and geometric relationships. We review them briefly and explain why none is appropriate as such for our problem.

3.1. Approaches based on symmetry axes

Several works ([2],[3],[4],[5]) have used symmetry axes to segment objects into parts. After edge detection, the object symmetry axes are detected. The axes are usually computed by taking the midpoint between two black pixels, and by linking these midpoints. The result is a collection of axes in which some are relevant for part segmentation and some are not. Rules are then used to select the axes that possibly describe parts. The boundary that has given rise to a selected axis corresponds to the boundary of a part.

These approaches work well for objects that are lightly textured. However, in our application, heavily textured object may be processed. For heavily textured objects, these approaches give rise to a very large collection of axes (the textures give rise to axes) in which it is very difficult to select the relevant ones. Hence, for our application, these approaches are not appropriate.

3.2. Approaches based on convexity points

These approaches are illustrated by the work of [6]. The basic strategy in these approaches is to segment the object based on the dominant convexity points on its boundary. After edge detection, the boundary is first scanned to locate dominant convexity points. Then, these convexity points are paired to find the joints between the parts. Once the joints are found, the parts are defined by the boundary lying between joints.

These approaches are not suited for our application, because some convexity points may be absent due to gaps between edges. In addition, objects, which can self-occlude in some viewpoints, cannot be segmented invariably into the same group of parts. Some convexity points can appear and disappear according to the viewpoint and change the segmentation.

3.3. Approaches based on a minimum description length criterion

These approaches group contour boundaries based on a minimum description length (MDL) criterion. In the work of [7], ellipsoid parts are extracted. The MDL criterion ensures that the ellipsoids fit in the best possible way the data (pixels). The length of the description may be calculated in the following way for an edges image. The associated length is zero for parts of the ellipsoid that are on boundary pixels, and for parts of the ellipsoid that do not fit any boundary pixel, the length is increased by one for every pixel that does no fit. Therefore, the smaller the description length, the better the fit. In these approaches, the MDL criterion is optimized globally for the entire image to extract the best combination of parts.

The parts obtained with these approaches may not be interpreted as simple volumetric primitives, since a single ellipsoid may fit more then one projection of volumetric primitives. Hence, objects are harder to interpret and compare. In addition, ellipsoids may not fit concave parts. Therefore, these approaches lack the generality needed in our application.

3.4. Approaches based on primitives cycles

These approaches attempt to build closed cycles of primitives (example: circular arcs or straight line segments). The cycles are constructed in such a way that they correspond to parts. Rules are defined for validating the addition of a new primitive in the cycle. The work of [8] implements this approach. A graph of the primitives (the nodes) and of the junctions (the arcs) of the object is constructed based on proximity. From this graph, all the possible closed cycles are found, and those that do not contain other cycles are considered as parts. The work of Jacobs [9] differs by the use of a convexity criterion and a saliency criterion on the cycles.

These approaches are not suited for our application, since in images of real scenes, the junctions are not always detectable and the parts are not always convex.

3.5. Approaches based on perceptual grouping and geometric relationships

These last approaches group constant curvature primitives (CCPs) based on geometric relationships. They are inspired by works on perceptual grouping ([10],[11]). The spatial arrangements of the CCPs are studied, and those that fit some criteria based on geometric relationships are considered as parts. Examples are PARVO [12] and the system developed by Dickinson [13]. PARVO uses symmetry and junctions to group CCPs into parts. Junctions give the clues about the structure of the object and ensure robust grouping. Dickinson system's groups CCPs in many stages. First, arrangements of CCPs are searched. Then, these arrangements of CCPs are grouped into faces, the faces are grouped into aspects, and the aspects are grouped into geons. Probabilities are associated to each hypothesis.

The systems that implement this approach, uses either junctions, or catalogue of arrangements of CCPs. Since, images of real scene are processed in our application, junctions are hard to detect. Hence, they cannot be used in the same fashion as in PARVO. Furthermore, catalogue of arrangements of CCPs are not flexible enough for our application since some arrangements of CCPs that are not present in the catalogue may exist in images.

Although the existing implementations of these approaches are not suited for our application, the principle behind them can be adapted to fit its needs. The next section explains why, and how this can be done.

4. Principle of our method

As mentioned previously, our application implies the processing of images of objects in real scenes. Grouping constant curvature primitives (CCPs) by geometric relationships is valid in this context, if the CCPs are grouped in such a way that the influence of texture and noise on the results is reduced. This can be done using structural information about the object. A number of high-level features convey structural information. In PARVO, the junctions are used as this high-level feature. However, junctions are hard to detect in images of real scenes. Catalogues of arrangements of CCPs could be used to obtain groups that respect given structures. However, this is too restrictive because it limits the type of arrangements of CCPs that can be processed. A feature easier to detect, more convenient and not restrictive should be used. For this purpose, the outline of the object has been chosen. Although it conveys less information than junctions do, it does not restrict the types of arrangements of CCPs that can be processed, and it simplicity allows a user to make a query by sketching only the outline of an object, or of its constitutive parts.

The principle of our method for part segmentation is to group two CCPs with a criterion based on intermediate-level geometric relationships and with the outline of the object to account for its structure. Subsequently, the complete boundary of the part is detected by making cycles with object CCPs to link the two grouped CCPs. The shape of the parts extracted is only limited by the structure of the object (obtained via the outline) and by the geometric relationships between the two CCPs grouped. Note that the outline is extracted using the method of [14] and the CCPs are extracted using MAGNO ([15]).

This method has the advantage of being able to extract any type of parts that is made of two compatible CCPs (as defined by the geometric relationship based criterion), along with their detailed shape. Therefore, the parts extracted may be non-convex. Furthermore, since the parts are made of two main compatible CCPs, they can be easily interpreted as simple volumetric primitives. Using the definition of generalized cylinders, the two compatible CCPs are the projected envelope of the region swept by the section, and the CCPs used to link the two compatible CCPs describe the projected section. In addition, the use of the outline reduces the complexity of the search for two compatible CCPs.

5. Details of part segmentation process

Let us now detail the segmentation process. The solution space of our part segmentation method is the set of all the possible CCPs pairs. The goal of our method is to select a subset of pairs of CCPs among these, which correspond to parts. This is done using a criterion that increases the probability that the selected pairs of CCPs correspond to the projected envelopes of the simple volumetric primitives composing the object. Once two compatible CCPs of a part are found and validated, the boundary of the part is completed by adding the CCPs that correspond to the section.

Since not all parts are on the outline of the object, part segmentation is realized in two stages. Stage 1 detects part with the help of the outline and intermediate-level geometric relationships. The parts are detected with the highest confidence at this stage. Stage 2 relies solely on intermediate-level geometric relationships. Part segmentation consider for grouping only CCPs that are on the outline, or inside the outline. The geometric relationships based criterion used for grouping is described in the next section, and then the two stages are detailed.

5.1. Geometric relationships based criterion

A unique criterion based on geometric relationships (CBGR) is used for both stages. This grouping criterion is based on five intermediate-level geometric relationships between a pair of constant curvature primitives (CCPs). They are the parallelism (P), the distance between the CCPs (D), the similitude in length (SL), the similitude in type (ST) and the overlap (O). These relationships are considered as intermediate-level, because they are computed at the CCP level. The grouping criterion is formulated as follows:

$$CBGR = \frac{P + D + SL + ST + O}{5} \qquad \text{(EQ .1)}$$

The next subsections show how the five intermediate-level geometric relationships are defined.

5.1.1. Parallelism (P)

Parallelism is in general a non-accidental property of the projected envelopes of simple volumetric primitives. This geometric relationship is the component of the grouping criterion used to support the grouping of two CCPs that are as close as possible of being parallel. The parallelism is computed in a similar fashion for both straight line segments and circular arcs. Hence, the parallelism is calculated by approximating the circular arcs by straight line segments passing by their endpoints. Defining the straight line segments as vectors pointing in an arbitrary chosen direction, parallelism is given by,

$$P = \left| \vec{a} . \vec{b} \right| \qquad \text{(EQ. 2)}$$

where $\vec{a}$ and $\vec{b}$ are two vectors representing straight line segments. The values obtained for P are between 0 and 1.

5.1.2. Distance between the CCPs (D)

The distance between the CCPs is used to ensure that the two compatible CCPs of a part are as close as possible. This geometric relationship is part of the grouping criterion because two close CCPs are more likely to define the boundary of the projected envelope of a single simple volumetric primitive. The distance between two CCPs is defined by,

$$D = 1 - \frac{\left\| mpCCP_1 - mpCCP_2 \right\|}{\max(\dim x, \dim y)} \qquad \text{(EQ. 3)}$$

where $mpCCP_1$ and $mpCCP_2$ are the midpoints of the two CCPs, and dim x and dim y are the number of columns and rows of the image. This value is between 0 and 1.

5.1.3. Similitude in length (SL)

The similitude in length is used to support the grouping of two compatibles CCPs that have about the same length. This is because in general, the two CCPs defining the projected envelope of simple volumetric primitives have similar lengths. For the similitude in length, the circular arcs are approximated by straight line segments in a similar fashion as for the parallelism. The actual length of the circular arcs is not used because the similitude in the distance between the endpoints is the property of interest. Indeed, we are interested in obtaining regions that are as rectangular as possible. The similitude in length is calculated as follow:

$$SL = \frac{\min(l_1, l_2)}{\max(l_1, l_2)} \qquad \text{(EQ. 4)}$$

where l_1 and l_2 are the length of the CCPs. This value is between 0 and 1.

5.1.4. Similitude in type (ST)

The similitude in type favours the grouping of CCPs of the same type (circular arcs or straight line segments). This is justified by the fact that, in general, simple volumetric primitives have their envelope defined by two CCPs of the same type. The similitude in type is computed by comparing the type of the two CCPs. The similitude in type is given by,

$$ST = \begin{cases} 1 \text{ if } Type(CCP_1) = Type(CCP_2) \\ 0 \text{ if } Type(CPP_1) \neq Type(CCP_2) \end{cases} \qquad \text{(EQ. 5)}$$

where *Type()* is an operator that returns the type of a CCP, and CCP_1 and CCP_2 are the two tested CCPs. This value is between 0 and 1.

5.1.5. Overlap (O)

For simple volumetric primitives, the two CCPs defining their envelopes are in general overlapping. The overlap relationship component of the grouping criterion supports the pairs of CCPs that are overlapping. The overlap (O) for a pair of straight line segments is defined as the length of the projection of CCP_2 on CCP_1 (Lo_1) and the length of the projection of CCP_1 on CCP_2 (Lo_2) divided by the sum of the length (L_1) of CCP_1 and of the length (L_2) of CCP_2. That is:

$$O = \frac{Lo_1 + Lo_2}{L_1 + L_2} \qquad \text{(EQ. 6)}$$

This value is also between 0 and 1. For circular arcs, the overlap is calculated by approximating them by straight line segments as for the parallelism.

For the two dark CCPs of Figure 2 (Group X), the value of the grouping criterion is 0.98, and the values of its components are P=0.99, D=0.94, SL=0.99, ST=1, and O=0.99. For the two dark dotted CCPs of Figure 2 (Group Z), the value of the grouping criterion is 0.34, and the values of its component are P=0.03, D=0.56, SL=0.10, ST=1, and O=0.

Figure 2

5.2. Part segmentation

Now that the grouping criterion has been defined, part segmentation itself can be described. Figure 2 illustrates the part segmentation process. The segmentation begins by first removing the CCPs that are outside the outline of the object (Figure 2a). Then, grouping attempts that take into account the segmentation stage are made (Figure 2b). The set of possible groupings depends on whether we are at Stage 1 or 2 as explained below. If a valid group is found, the boundary of the part is completed (Figure 2c). If the segmentation process is at Stage 1, the CCPs making out the part are removed (Figure 2d). The segmentation process cycles between the three last steps (Figure 2c,d and e) until no more groups can be found. The segmentation process is completed (Figure 2f). The next subsections describe in details CCPs grouping and group validation (Figure 2b), boundary completion (Figure 2c), part CCPs removal (Figure 2d) and the difference between the two segmentation stages.

5.2.1. CCPs grouping

Grouping attempts are made from all the available constant curvature primitives (CCPs). The availability of some CCPs depends on the segmentation process stage. At Stage 1, only the outline CCPs are available for grouping, whereas at Stage 2, all CCPs are available. CCPs grouping first start by selecting the longest CCP available. To select the longest CCP, cocurvilignity (colinearity or cocircularity) is used to reconstruct fragmented CCPs. A reconstructed fragmented CCP or a CCP that does not need reconstruction is called a reconstructed CCP (RCCP). For each grouping attempts, the longest RCCP available (seed RCCP) is tentatively grouped with all the RCCPs available. The quality of each grouping attempts is measured by the value of the grouping criterion. The pair of RCCPs that gets the highest value for the grouping criterion, which is also higher then the grouping criterion threshold, is considered as a possible group. The grouping criterion threshold is the minimum grouping criterion value considered as reflecting a valid group. This group is then validated.

Figure 3

5.2.2. Group validation

Group validation ensures that the groups that are made are the ones in which we have more confidence. The groups in which we have more confidence are groups built from outline RCCPs, and groups that bound the object. For groups made of two outline RCCPs, the impact of the interior RCCPs on these groups must be investigated. This is the goal of Verification 1 of the group validation step. Figure 3 shows the three possible situations in which a group of two outline RCCPs can be formed. Although the two RCCPs respect the grouping criterion, they do not necessarily correspond to the best group (see Figure 3, Situation A). Verification 1 determines to which situation a grouping corresponds and reject or accept the group accordingly. Groupings that correspond to Situations B and C (see Figure 3) are accepted. Groupings that correspond to Situation A are rejected. The grouping situation is established by reattempting to group the outline RCCPs under validation with interior RCCPs. If no group is found, the group under validation corresponds to a grouping in Situation C. If the first two RCCPs under validation are grouped with the same interior RCCPs, the group under validation corresponds to a grouping in Situation B. In any other cases, the group under validation corresponds to a grouping in Situation A. Verification 1 applies only to groups made during Stage 1.

A match may not be valid because the pair of matched RCCPs bounds a portion of the scene background. Verification 2 of the group validation step addresses this issue. To verify if the inside area of the object is between the two matched RCCPs as it should, coordinates of points, sampled between the two RCCPs, are verified to confirm if they belong to the set of coordinates of points of the inside area. Since the outline of the object is known, the area it covers is known and so is the set of points inside this area. Figure 4 shows how the sampled points are chosen. If all the sampled points belong to the set of points of the inside area, the two RCCPs bound a region of the object, and hence, the grouping is accepted under this verification. This verification applies to both stages.

Figure 4

If a grouping is rejected for one of the two verifications, this pair of RCCPs is removed from the solution space along with all the pairs made of the current seed RCCP. A new grouping attempt is initiated with the next longest RCCP available.

5.2.3. Boundary completion

When a valid group is found, the complete boundary of the part is constructed from the set of CCPs of the object or from virtual straight line segments. This is done by attempting to make a path of CCPs that links the endpoints of the two grouped RCCPs. The paths are constructed based on two criteria. First, all the CCPs of a given path must remain inside the area defined by the lines passing by the endpoints of each of the two grouped RCCPs. The dotted path in Figure 2 (Path Y) is invalid because of this criterion. Next, two consecutives CCPs in the path must be oriented as much as possible in the same way. The path is constructed using proximity between endpoints.

To add a CCP in the path, a set of nearby CCPs is first built. Then, the orientation criterion is applied to select the CCP to add to the path. RCCPs can be added to the path if they reach directly the destination RCCP (the RCCP to reach to complete the path), and if they bridge a gap. During the construction of the path, there is no backtracking. Hence, if the path does not reach the destination RCCP, the path construction fails. In this case, the two unlinked endpoints are linked by a virtual straight line segment. If two virtual straight line segments are required to link the endpoints, they are added so that they do not intersect and are both as short as possible. The complete boundary obtained is considered as the boundary of the projection of a part.

5.2.4. Part CCPs removal

For parts obtained at Stage 1, the CCPs that are inside the area of the part, and the RCCPs or CCPs that define its boundary are removed from the solution space (i.e. they are made unavailable).

5.2.5. Differences between the two segmentation stages

The two segmentation stages differ mainly by the set of available RCCPs for grouping. At stage 1, only the outline RCCPs are available. For stage 2, all the RCCPs are available, but groupings are attempted first with outline RCCPs. For the second stage, the outline RCCPs are still considered more significant. Furthermore, for Stage 2, only Verification 2 of the group validation step is made because no structural information is available to perform verification 1. In addition, at Stage 2, the part CCPs are not removed. This is because no structural information is available and hence, the groups are less certain. We rather keep all the ungrouped RCCPs, consider all the possibilities, and obtain overlapping parts, than make erroneous premature decisions. These overlapping parts will have to be further processed to determine the best ones. Note, however, that the seed RCCP is made unavailable in both stages to avoid redundancy. The segmentation proceeds to Stage 2 when no more groups can be formed at Stage 1.

6. Results and analysis

To validate our approach, three experiments have been designed, each investigating a specific aspect. The first experiment verifies the validity of our grouping criterion by processing synthetic constant curvature primitives (CCPs) drawings. The second experiment extends the first experiment, by verifying the validity of our grouping criterion for images of real scenes. The third experiment investigates the consistency of the segmentation for several images of the same object. The next subsections present each of these experiments, followed by a discussion about the parameters involved in the segmentation process and computation times.

Figure 5

6.1. Experiment on synthetic CCPs drawings

The CCPs drawings chosen to validate our algorithms depict objects made of simple volumetric primitives, likes cylinders, prisms, cones, etc (see Figure 5). Note that these same drawings were used to demonstrate the capabilities of PARVO [12]. Ideally, the parts obtained by our method should correspond to each of the individual simple volumetric primitives composing the object, if their projections are bound by at least two outline CCPs. For projections bounded by less then two outline CCPs, the individual faces of the volumetric primitives should be found. Our experiments show that it is the case for twenty-one of the twenty-three CCPs drawings tested. The segmentation obtained for the drawings of a stepladder is shown in Figure 6. Note that the

Figure 6

parts labelled A and B correspond to the two visible faces of the projected prism. The prism is not extracted as a whole because it has no CCPs on the outline.

Figure 7 shows the results for the two CCPs drawings that were not segmented correctly. For the pen, the reason is the failure to reconstruct a fragmented circular arc (see the fragmented circular arc indicated by two arrows in Figure 7). This reconstruction failure causes the under segmentation of the body of the pen and the clip. Note that for this view of the pen, the clip could not be segmented from the body even if there was no reconstruction failure because it is almost completely inside the body. The area of the clip inside the body is considered as texture by our method. The remaining CCPs of the clip could not form any group, and hence it could not be segmented. For the briefcase, the handle, which is made of volumetric primitives that are not bounded by the outline, is not properly segmented because of the grouping criterion. To obtain a good result for this CCPs drawing, the similitude in length and the distance between the CCPs should not have the same weight in the grouping criterion for different region of the handle. Since a unique grouping criterion is used for the entire image, it is not possible to perfectly segment the handle. This problem is occurring because the handle is made of small CCPs of about the same length that are nearby.

Figure 7

This experiment demonstrates that our segmentation method can process synthetic CCPs drawing exactly as expected for 91,3% of the drawing we have tested. For the drawings that were not entirely processed correctly, the problems causing the errors are well understood, and affect only local portions of the objects. Therefore, our method is suited to segment objects and can be further validated with images of real scenes to observe its performance in presence of noise, textures, and shadows.

Figure 8

6.2. Experiment on images of real scene

The goal of this experiment is to show that our grouping criterion and our method can be used to process images of an object in a real scene. For this purpose, twenty-four images of objects in real scenes have been processed. Among these, nineteen were segmented properly. Figure 8 shows the part segmentation result obtained for the image of a compass. The partial ring at the top of the compass is extracted as a disc, because circular arcs describing the ring are missing in this region. The method has performed adequately with the CCPs available in this region.

Although the results obtained are not perfect for all the images tested, the majority of the parts obtained correspond to simple volumetric primitives making out the object. Hence, in general, the method reach its goal, which is to segment an object into its simple volumetric primitives, as it, succeeds for 79.2% of the images. For the images that are not segmented correctly, the errors are only local to a portion of the object. The segmentation method handles relatively well textures and noise as fragmented CCPs are reconstructed and the outline is used to obtain clues about the structure of the object. The following paragraphs discuss some segmentation errors.

Figure 9

Figure 9 shows a result for an airplane with segmentation errors. Parts corresponding to wings (Part A, C, D, F, G, H, J, K) and an engine (Part I) have been segmented correctly. The fuselage is over segmented (part B and E). This is because the reconstruction of the fragmented boundary of the fuselage has failed. There is a spurious background part (part L). This is due to a failure of the object detection step. Therefore, the presence of this part is not a failure due to the grouping criterion. Finally, there are parts (part M, N, O) that correspond to markings on the airplane. These parts are found because they are in between two simple volumetric primitives. In this case, our grouping criterion behaved correctly in the circumstances. The CCPs composing these parts were available for grouping, and they have been grouped as it should. However, these parts are irrelevant for the modelling of the structure of this particular airplane. In this case, they could be removed by comparing their size with the size of the other part of the airplane and by analyzing their location (they are in between two nearby parts).

Figure 10

Figure 10 shows the result from another image with some segmentation errors. Aside from a leg that has been over segmented (part E, G) because two very close parallel straight line segments were in this area, two legs (part B) and two rungs (part F) are each extracted as one part. This is because of the viewpoint. The two individual parts become overlapped, and they appear as a prism for which two faces are seen. Since, CCPs of the two individual parts are on the outline, are mutually grouped, and are validated by the Verification 2 of the group validation step, they are extracted as a whole. Although these are not desirable parts, our segmentation algorithm is working correctly. Higher-level reasoning is needed to prevent this kind of grouping. This shows that our segmentation method is not completely independent of the viewpoint. However, for our image query application, a total independence from viewpoints is desired, but not essential. Obtaining consistency of description for a wider range of viewpoints than is possible with 2D contour models was the intent for the design of this segmentation method. We believe that description by simple volumetric parts allows better viewpoint independency performance when comparing 2D images of objects in a database. The third experiment will investigate this consistency of description.

Figure 11

6.3. Experiment on segmentation consistency

For modelling an object by a graph of parts, a method that can segment object into parts is not enough. That method must give consistent segmentations for different images of the same object and for a certain range of viewpoints as noted in the previous section. This issue is not addressed in the majority previous works on part-based description of objects. This experiment verifies whether our segmentation method gives consistent segmentations. Fifteen images of a desk lamp have been processed for this experiment. They are shown in Figure 11. Two types of segmentation are obtained (see Figure 11). The segmentation of the lamp pole in one or more parts is not considered as a difference in the type of segmentation since they can be easily merged into one (see [1]). The difference in the types of the segmentation for the fifteen images is in the segmentation of the lampshade. For fourteen of the fifteen images, it is segmented as one part, and for the remaining image, it is segmented as two parts. The lampshade is segmented as one part for the majority of the images, because the circular arc marked with an X in Figure 11 is the longest CCPs of the lampshade. Hence, it is grouped first, and it gives a valid group with the opposite circular arc. For the image that does not give this segmentation, the circular arc marked with an X is still tentatively grouped with the opposite circular arc, but this group is not valid under Verification 1 of the group validation step. This is because there is an additional circular arc between these two that correspond to the light bulb visible under the lampshade.

This experiment shows that our method is capable, in general, of giving consistent segmentation for different images of the same object. The views of the lamp used in this experiment are quite varied, and the 2D contour shapes of the lamp quite different. Obviously, texture (like the light bulb) can cause different segmentation. However, these segmentation errors remain local.

6.4. Segmentation parameters and computation times

To segment images with our method, three parameters may be adjusted. They all involve the reconstruction of CCPs. They are the cocircularity tolerance, the colinearity tolerance, and the gap accepted between two CCPs forming a reconstructed CCP. The cocircularity and colinearity tolerances concern the perpendicular distance between two circular arcs or two straight line segments having the same orientation. These tolerance parameters adjust the threshold in pixels of the maximum perpendicular distance allowed. The threshold on the gap is expressed as the percentage of object CCPs support for a reconstructed CCP. If these thresholds are too high, any two CCPs can form a reconstructed CCP. If the thresholds are too low, then fragmented CCPs cannot be reconstructed. For synthetic line drawings, unique values can be used for these parameters. As expected, for images of real scenes, these parameters vary from one image to another, although the range of values to obtain identical segmentations overlap for several images. This is particularly the case for the gap and colinearity parameters (Table 1). We believe that the selection of the thresholds could be automated, using constraints on the number of parts and the number of overlaps, and by analyzing the variation in the dimension of parts in specific areas when the threshold values are changed.

Table 1

Another important issue is the computation time. In the context of a database query engine, the query image must be processed rapidly. Without optimization, the computation times obtained on an Athlon Thunderbird 1.2 GHz are all under 10 seconds (see Table 1). The complexity of the method is in the order of n^2, where n is the number of CCPs. Note, that each time a group is found, CCPs are removed from the solution space.

7. Conclusion

This paper has presented a new method to segment an object from an image of a real scene into parts that correspond to the projections of simple volumetric primitives. These projections of simple volumetric primitives can be described mainly by their two main sides. They are obtained by grouping two constant curvature primitives (CCPs) that respect a criterion involving simple geometric relationships, like symmetry, length, distance, similarity and overlap. The outline of the object is also used to add to the grouping process knowledge about the structure of the object.

The results obtained show that our method performs well enough to be used in the context of our image database query engine. In general, the segmentations obtained for different images of the same object are consistent. That is, for the same object, the segmentation is usually the same for a range of viewpoints that are not too odd. The method is capable of segmenting objects made of well-defined simple volumetric primitives, like cylinder and prism. It can also tolerate a certain amount of marking and texture on the object.

Future work will consist of testing these results to query object in a database. First, qualitative volumetric primitives will be inferred from these projections of simple volumetric primitives and a qualitative volumetric primitives graph of the object will be made. Then, a method to compare these graphs will be implemented. Finally, the performance of the complete system will be characterized by using the query engine on a variety of images. Currently, all these tasks are under design and implementation (see [1], [16]).

Acknowledgements

This work is supported by a postgraduate scholarship from the Natural Sciences and Engineering Research Council of Canada (NSERC) and from le Fonds pour la Formation de Chercheurs et l'Aide à la Recherche (FCAR).

References

[1] Bilodeau, G.A. and R. Bergevin. Generic Modeling of 3D Objects from Single 2D Images. in International Conference on Pattern Recognition. 2000. Barcelona, Spain. pp. 770-773.

[2] Tek, H., P.A. Stoll, and B. Kimia. Shocks From Images: Propagation of Orientation Elements. in Computer Vision and Pattern Recognition. 1997. Puerto Rico. pp. 839-845.

[3] Siddiqi, K. and B.J. Kimia, Parts of Visual Form: Computational Aspects. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1995. **17**(3): pp. 239-251.

[4] Kelly, M.F. and M.D. Levine. Region-Based Grouping Operations for Locating and Describing Objects. in Vision Interface'95. 1995. Québec, Canada.

[5] Zerroug, M. and R. Nevatia, Volumetric Descriptions from a Single Intensity Image. International Journal of Computer Vision, 1996. **20(1/2)**: pp. 11-42.

[6] Bennamoun, M. and B. Boashash, A Structural-Description-Based Vision System for Automatic Object Recognition. IEEE Transaction on Systems, Man and Cybernetics - Part B: Cybernetics, 1997. **27**(6): pp. 893-906.

[7] Pilu, M. and R.B. Fisher, Model-Driven Grouping and Recognition of Generic Object Parts from Single Images. Robotics and Autonomous Systems, 1997(21): pp. 107-122.

[8] Jacot-Descombes, A. and T. Pun, Asynchronous Perceptual Grouping: From Contours to Relevant 2-D Structures. Computer Vision and Image Understanding, 1997. **66**(1): pp. 1-24.

[9] Jacobs, D. Robust and Efficient Detection of Convex Groups. in Computer Vision and Pattern Recognition. 1993. pp. 770-?

[10] Biederman, I., Human Image Understanding: Recent Research and a Theory. Computer Vision Graphics Image Processing, 1985. **32**: pp. 29-73.

[11] Lowe, D.G., Perceptual Organization and Visual Recognition. 1985: Kluwer Academic Publishers.

[12] Bergevin, R. and M.D. Levine, Generic Object Recognition: Building and Matching Coarse Descriptions from Line Drawings. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1993. **15**(1): pp. 19-36.

[13] Dickinson, S.J., A.P. Pentland, and A. Rosenfled, 3-D Shape Recovery Using Distributed Aspect Matching. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1992. **14**(2): pp. 174-198.

[14] Bilodeau, G.A. and R. Bergevin. Outline-based Part Segmentation using Intermediate-level Symetries. in Vision Interface'99. 1999. Trois-Rivières, Canada. pp. 66-73.

[15] Moktari, M. and R. Bergevin. Multiscale Compression of Planar Curves using Constant Curvature Segments. in 14th International Conference on Pattern Recognition. 1998. Brisbane, Australia. pp. 744-746.

[16] Bilodeau, G.A. and R. Bergevin. Modeling of 2D Parts Applied to Database Query. in Vision interface 2001. 2001. Ottawa, Canada. pp. 228-235.

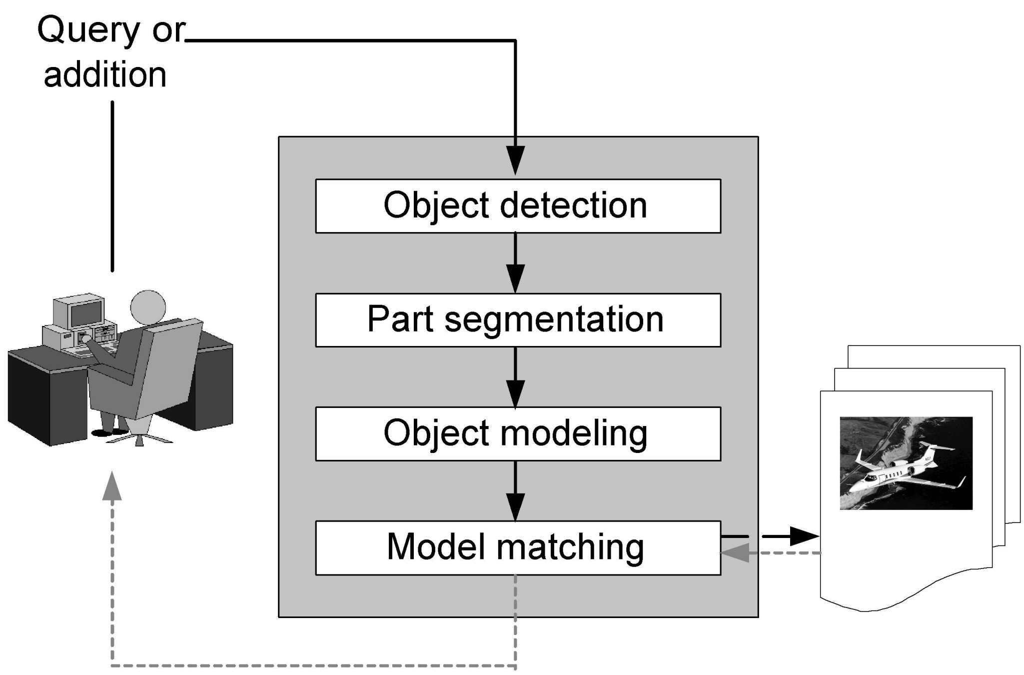
Query or
addition
Object detection
Part segmentation
Object modeling
Model matching

Figure 1

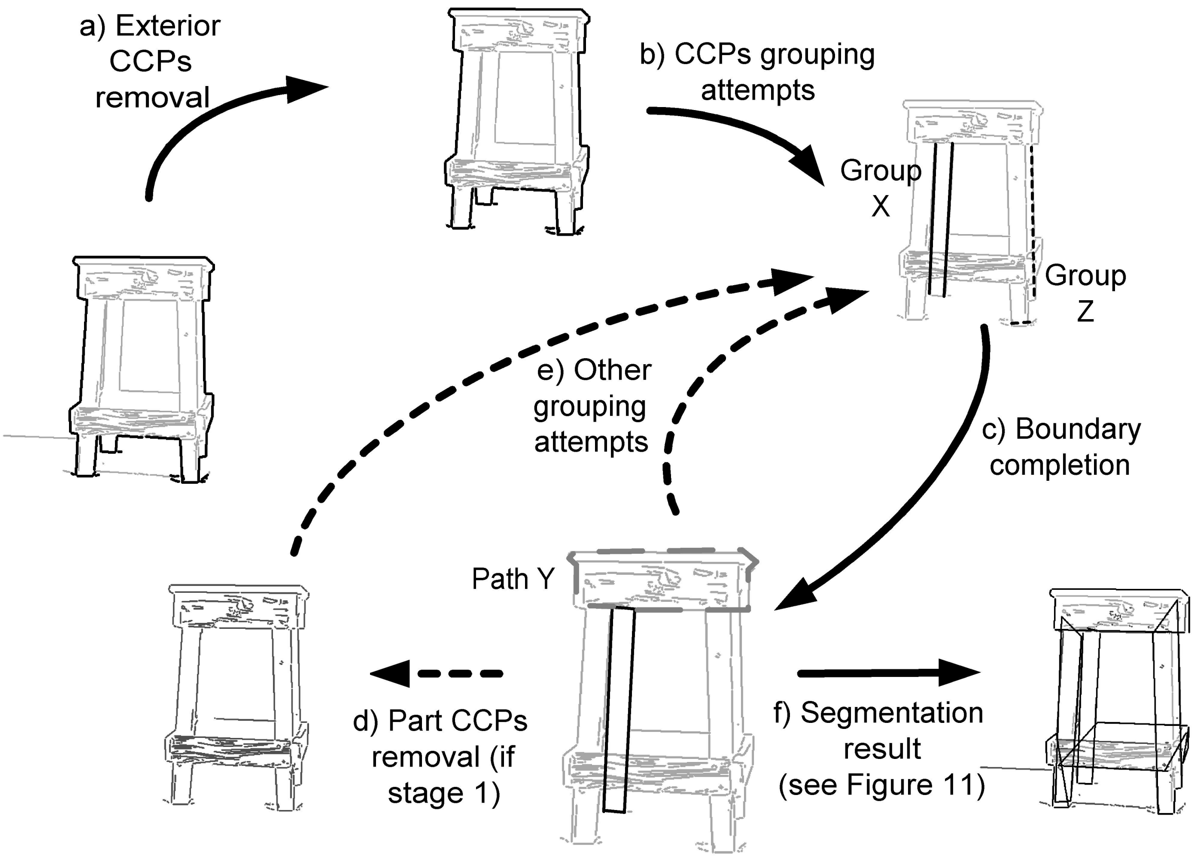
a) Exterior CCPs removal
b) CCPs grouping attempts
Group X
Group Z
c) Boundary completion
e) Other grouping attempts
Path Y
d) Part CCPs removal (if stage 1)
f) Segmentation result (see Figure 11)

Figure 2

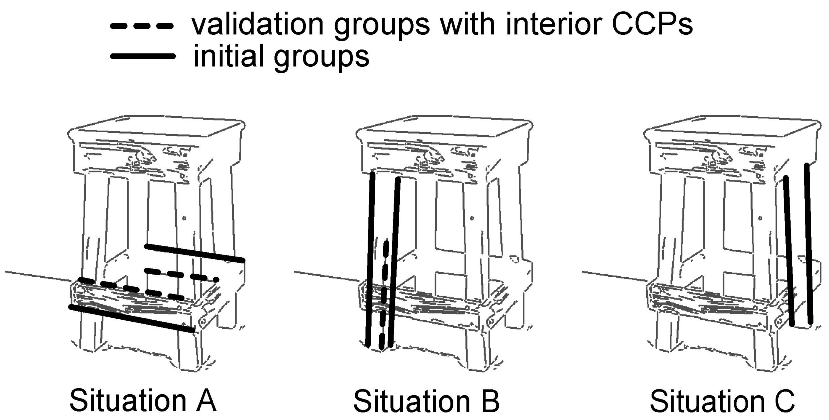
validation groups with interior CCPs
initial groups
Situation A
Situation B
Situation C

Figure 3

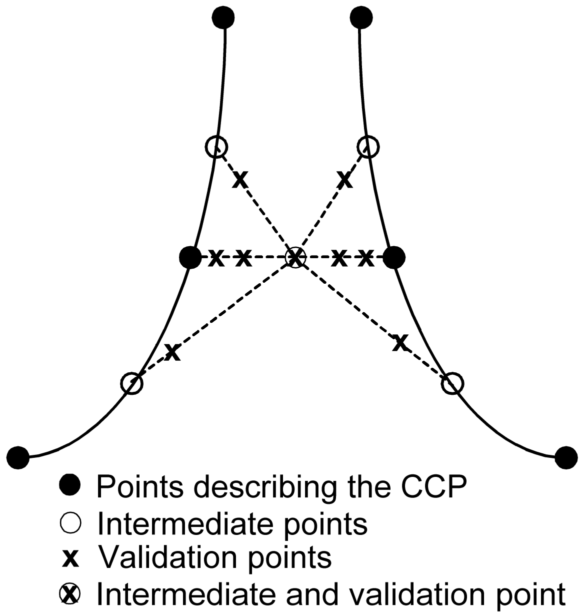
Points describing the CCP
Intermediate points
x Validation points
Intermediate and validation point

Figure 4

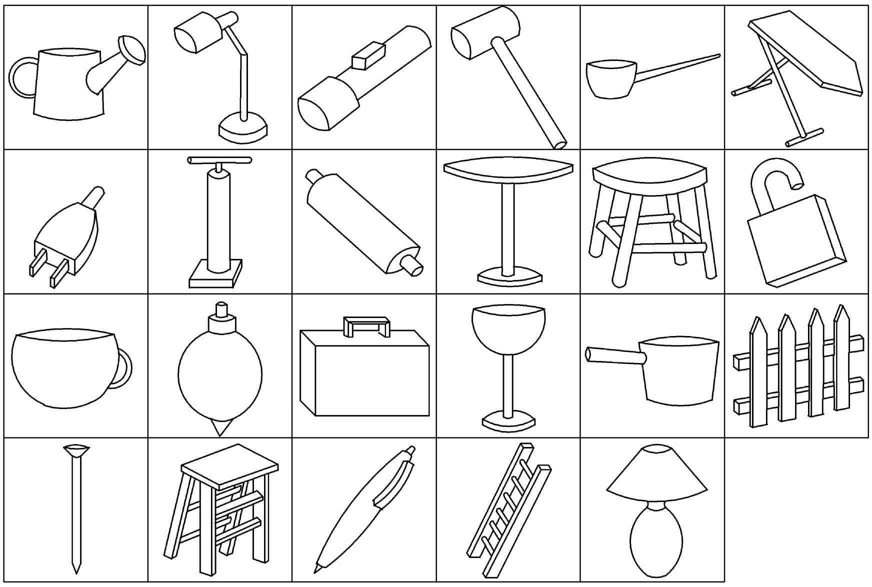

Figure 5

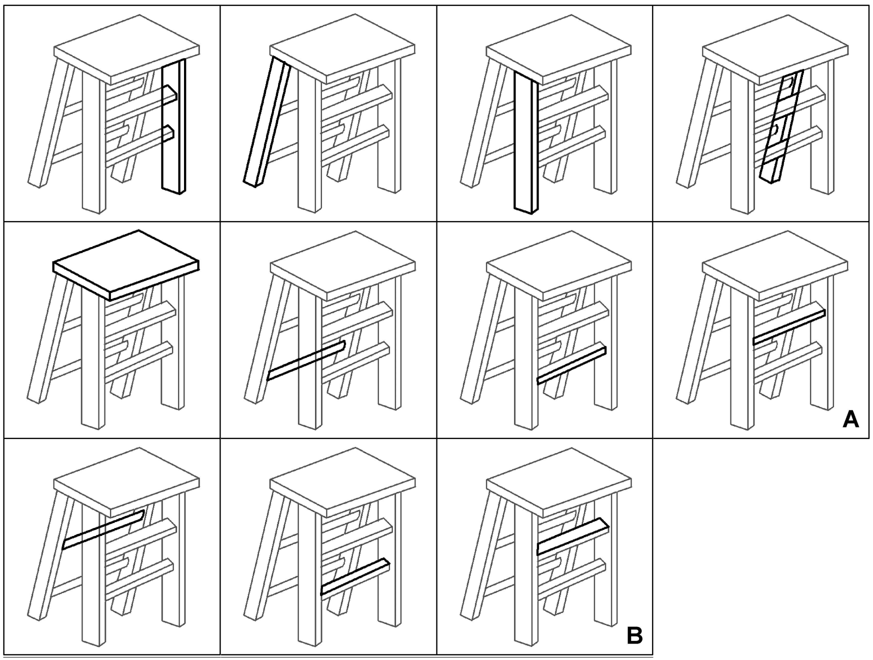

Figure 6

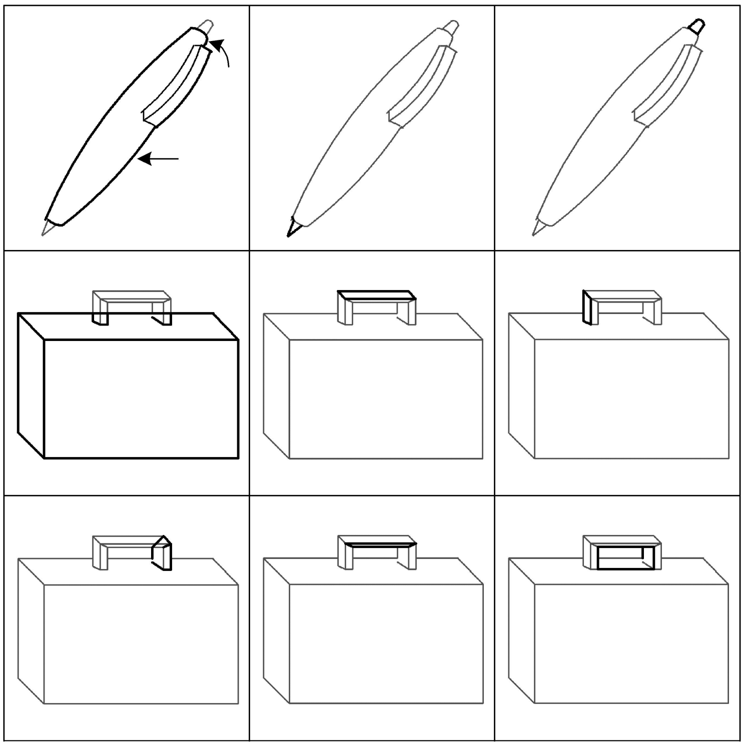

Figure 7

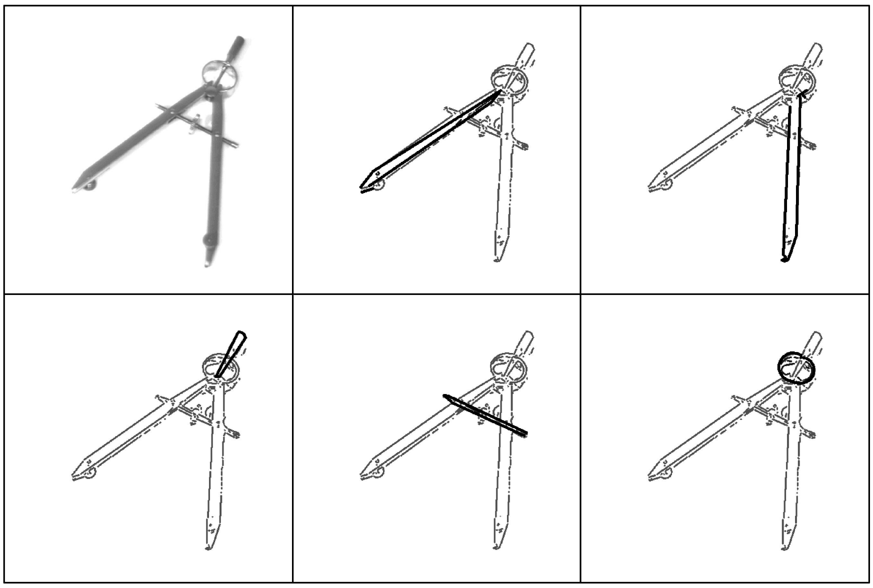

Figure 8

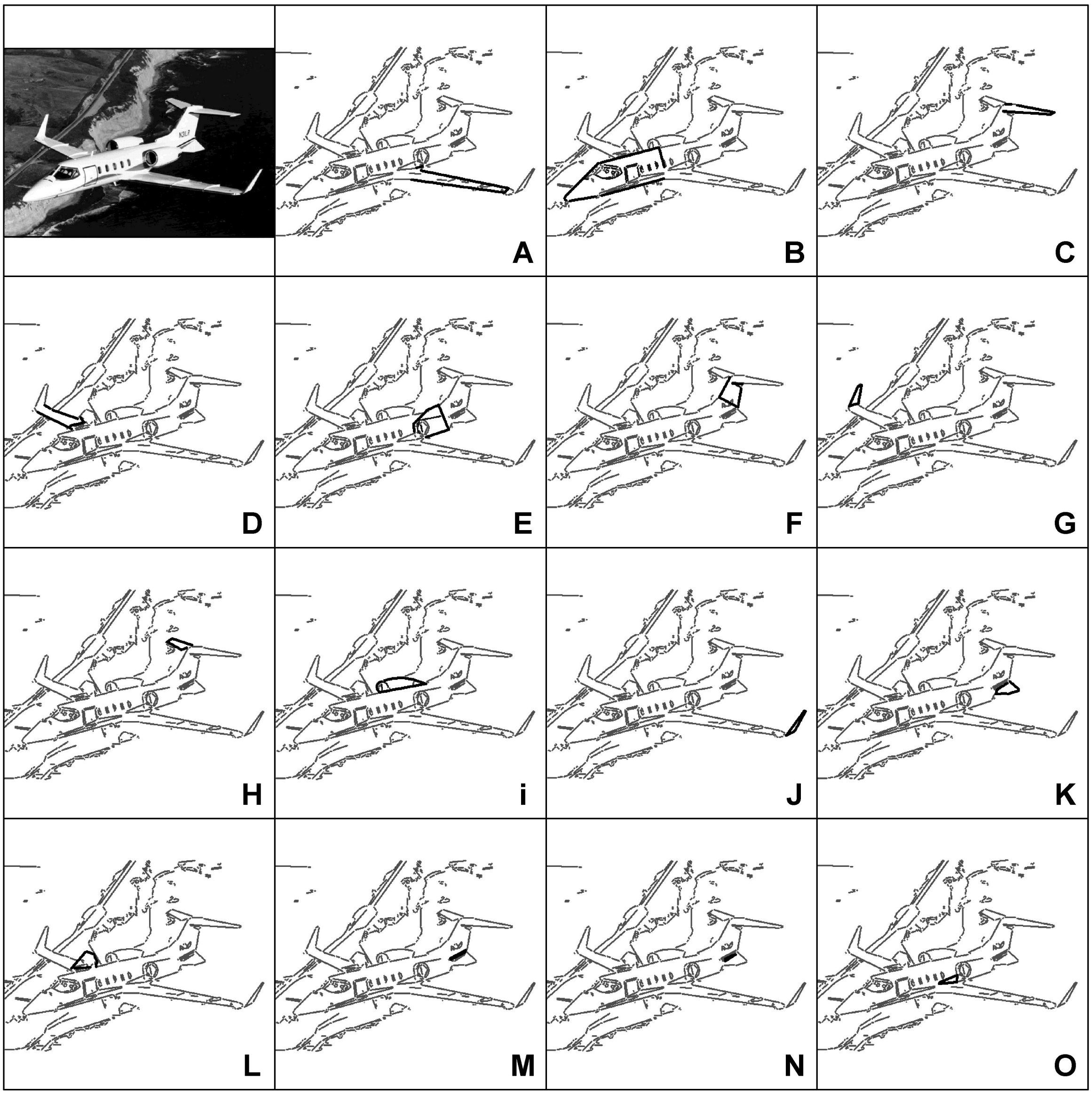

Figure 9

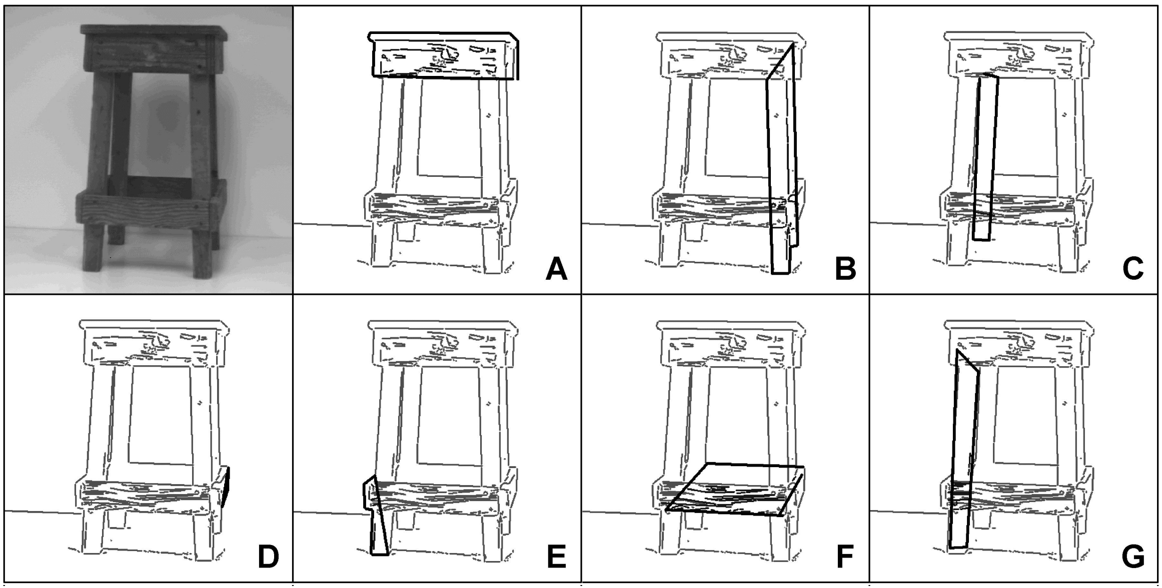

Figure 10

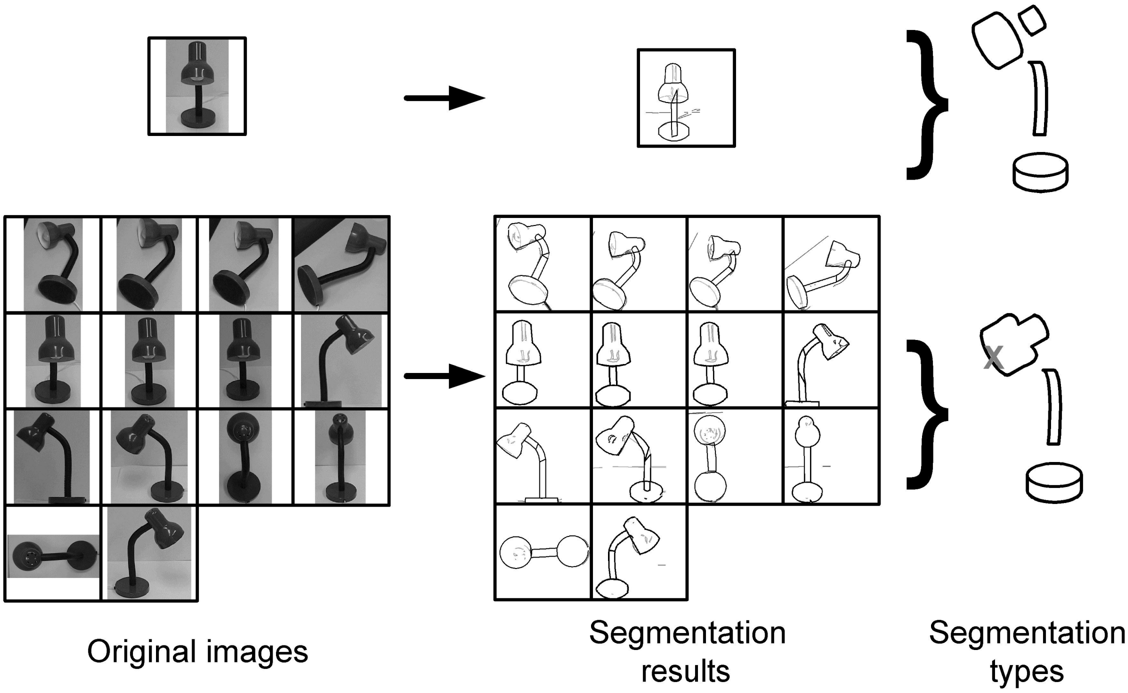
Original images
Segmentation results
Segmentation types

Figure 11

Figure 1: Overview of the query engine.

Figure 2: The segmentation process

a) The segmentation begins by removing exterior CCPs, b) CCPs grouping attempts are made, c) The boundary of the part is completed, d) The CCPs making the part are removed (if Stage 1), e) Other grouping attempts are made, f) Segmentation result.

Figure 3: Validation with interior CCPs

For Situation A, two parts with only one CCP on the outline are grouped. If both CCPs are re-grouped with interior CCPs, the group becomes invalid since they are both best grouped with other CCPs. For Situation B and Situation C, the groups are validated because both grouped CCPs are grouped with the same interior CCP or they are again grouped together.

Figure 4: Verifying if the object is between to CCPs.

Intermediate points are midpoint between CCPs points. Validation points are midpoint between intermediate points or between an intermediate point and the midpoint of a CCP.

Figure 5: The line drawing used for the first experiment.

Figure 6: Segmentation result for a stepladder.

Figure 7: Segmentation results for two problematic drawings

Figure 8: Segmentation result for a compass image.

Figure 9: Segmentation result for an airplane image.

Figure 10: Segmentation result for a stool image.

Figure 11: Two types of segmentation for images of the same desk lamp.

Table 1: Parameters values to obtain identical segmentations and computation times of some images of real scenes.

Image name	Number of CCPs	Cocircularity tolerance (pixels)	Colinearity tolerance (pixels)	GAP (%)	Computation time (s)
Learjet1	171	[0,1]	3	[20,30]	7.14
Learjet2	70	[0,3]	[3,5]	[40,80]	1.64
Compass	92	4	[7,8]	[40,50]	2.90
Stool2	177	[0,4]	7	[40,50]	7.62
Cup	39	[4,5]	[1,20]	[40,50]	0.83
VR-Room_Lamp1	61	[1,3]	[1,14]	[20,50]	1.27
VR-Room_Lamp2	41	[6,11]	[1,20]	[20,30]	0.65
VR-Room_Lamp3	37	[3,6]	[1,20]	[20,40]	0.64

Summary

This paper presents one significant aspect of a project aimed at the design of an image database query engine, where the images are searched at the 3D object-level. This approach is a novelty since the majority of existing image database query engines search images by comparing the colours, the textures and the 2D shape of regions in the images. The main aspect and contribution of this paper is a new method for object part segmentation, from circular arcs and straight line segments primitives, that is suited to process images of object in real scene. In the context of this work, 2D parts are defined as regions delimited by groups of circular arcs and straight line segments (constant curvature primitives), which can later on be interpreted as the projections in the plan of simple volumetric primitives, likes cylinders and prisms. Our new method is based on perceptual grouping. It groups constant curvature primitives (CCPs) using intermediate-level geometric relationships and object outline information. The use of the outline of the object reduces the complexity of the grouping process, since many groups are discarded based on this information. The principle of our method for part segmentation is to group two CCPs with a criterion based on intermediate-level geometric relationships and with the outline of the object to account for its structure. Subsequently, the complete boundary of the part is detected by making cycles with object CCPs to link the two grouped CCPs. The shape of the parts extracted is only limited by the structure of the object (obtained via the outline) and by the geometric relationships between the two CCPs grouped. We will show that our approach is suited to process 3D objects in 2D images of real scenes by several validating experiments.

Biographical sketch

GUILLAUME-ALEXANDRE BILODEAU received the B.Sc.A. Degree in Computer Engineering from Université Laval, Canada, in 1997, and the M.Sc.A. Degree in Electrical Engineering from the same University in 1999. He is currently a Ph.D. candidate at the Computer Vision and Systems Laboratory of Université Laval. His research interests include object recognition, image database query and graph matching. He is a member of Quebec's association of professional engineers (OIQ), and a member of CIPPRS.

ROBERT BERGEVIN received the B.Eng. degree in Electrical Engineering and the M.A.Sc. Degree in Biomedical Engineering from École Polytechnique in Montréal, in 1982 and 1984, respectively, and the Ph.D. degree in Electrical Engineering from McGill University in 1990. From May 1984 to September 1985, he was a practicing clinical engineer at Sacré-Coeur Hospital in Montréal. From March to August 1990, he worked as a researcher in the field of automated inspection at the Production Engineering Research Laboratory of Hitachi Ltd. in Yokohama, Japan. His research interests encompass computer vision and robotics with an emphasis on object and scene description, modeling and recognition. Since November 1990, he has been with the Computer Vision and Systems Laboratory at Laval University, Quebec City, where he is currently a Full Professor in the Department of Electrical and Computer Engineering. Dr Bergevin is a member of the Province of Quebec's association of professional engineers (OIQ), a member of the IEEE Computer Society, and CIPPRS, and an Associate Editor of Pattern Recognition, the Journal of the Pattern Recognition Society.