Generic Detection of Multi-Part Objects by High-Level Analysis

Jean-François Bernier
Laval University
Computer Vision and Systems Laboratory
Quebec, Canada
jfbernie@gmail.com

Robert Bergevin
Laval University
Computer Vision and Systems Laboratory
Quebec, Canada
bergevin@gel.ulaval.ca

Abstract

A method is proposed to detect multi-part man-made or natural objects in complex images. It consists in first extracting simple curves and straight lines from the edge map. Then, a search tree is expanded by selecting and ordering the segmented primitives on the basis of generic local and global grouping criteria. The set of partial contours provided by the parallel search are combined into more complex forms. Global scores produce a sorted list of potential object silhouettes.

1. Introduction

Multi-part objects are everywhere, from living beings to man-made objects, rigid or deformable, articulated or not. They can be a person, with a head, body, two legs, and two arms, or an airplane, with its nose, body, two wings, and tail. Current work in detection of such objects in images is often too specific and it lacks efficiency and noise tolerance. A new fully deterministic and generic method is proposed whose goal is to come closer to the capacity of humans to detect interest regions in complex images of multi-part objects. Potential interest objects are located by orderly selecting contour primitives on their boundary, based on a limited set of simple grouping criteria common to all members of the abstract category of multi-part objects.

Existing studies in the field of shape detection are numerous but they fall short to satisfy our needs. Model-based object recognition approaches match local features to a predefined model in order to find the pose of a specific object [1]. Recently, new powerful techniques were proposed to learn local appearance features from examplar images and apply them to detect interest objects [3]. Despite some impressive results, they are still too limited in terms of needed viewpoint-invariance and genericity. Many techniques are based on more generic shape models but they usually deal only with relatively simple shapes or images [7]. Approaches based on interest points are popular and they may also bring information on the image contents [12]. However, they are not yet applied to generic shape extraction tasks. Region segmentation share similarities with our proposed method since in both cases the image is first segmented into pieces, before deciding which ones to fuse in order to reconstruct interest object, just like a puzzle. However, getting the meaningful pieces is more difficult with regions than contours, especially in terms of under-segmentation. Besides, shape-based grouping criteria are easier to define on a partial contour silhouette than on a partial region silhouette.

Contour-based perceptual grouping is an efficient preprocessing step in both specific and generic shape detection methods. However, few existing methods apply it fully up to the detection of generic shapes of a sufficient complexity [2]. For instance, [6] proposes a shape detection algorithm tolerant to missing and spurious points, but the main criterion is the convexity of the group. Concavities as well as convexities are essential features of multi-part objects. Inter-line affinity is computed in [7, 5]. Proximity, continuity and closing criteria are defined on that basis. The extracted contours are numerous, they may appear anywhere, and they may look anyhow since only local, or simple global [4], criteria are enforced. The great diversity of multi-part objects asks for a greater set of local and global criteria.

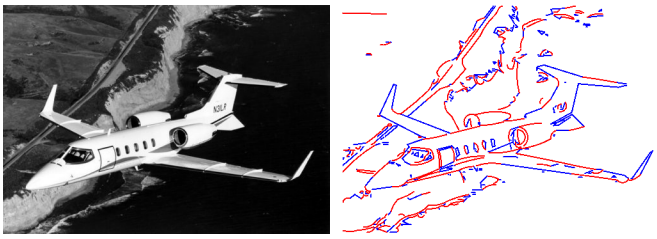
2. Basic concept

Figure 1(b) is a constant-curvature contour primitive (CCP) map of Figure 1(a) obtained using a custom segmentation algorithm [9]. A possible solution to our detection problem is any ordered group of CCPs from the map. Considering an average map of 400 CCPs and solutions with 30 CCPs, the number of possible solutions is about 10^{86}.

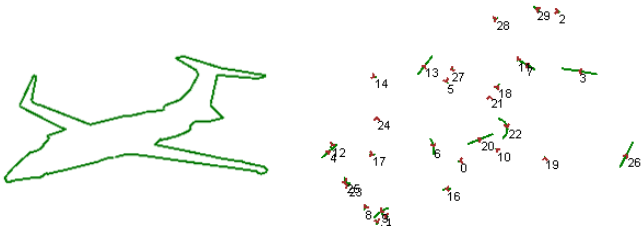
Figure 1(c) is the best solution as interactively selected by a human. It is referred to as SGT, for subjective ground truth. SGT is not know by the algorithm and will only serve

in assessing the quality of obtained results. Let us assume that an algorithm provides a scoring function for solutions. FGT, or formal ground truth, is the possible solution with highest score. FGT is usually not known either as it would require to generate all of a nearly infinite number of solutions and score each of them. More practically, a subset of the possible solutions is considered and the one with the highest score is selected. The selected optimal solution is only an approximation of FGT. It is referred to as FGTa.

FGTa may be the same as FGT, but this can seldom be verified in images of typical complexity. The goal of the proposed method is to generate, in an efficient manner, an FGTa as close as possible to SGT, for a variety of complex images of multi-part objects. Figure 1(d) shows an example of a random solution with 30 CCPs (each CCP is imaged with a small arrow and a number). In the proposed method, local grouping criteria help discard such a poor solution early on. Best retained solutions according to global criteria are to be similar to SGT (see Figure 6).



(a) Image (b) CCP map



(c) SGT (d) Random solution

Figure 1. Basic concept

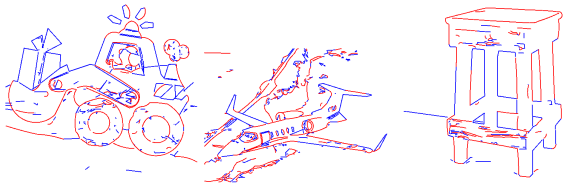
3. Input data

Images in Figure 2 are segmented to produce contour primitives maps shown in Figure 3. Different difficulties may arise from the segmentation step. For instance, conflicting continuous primitives, discontinuous primitives, incomplete primitives, and overlapping primitives. The proposed method is meant to address all of them.

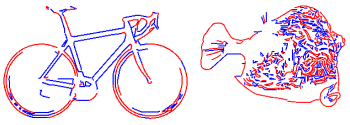
(a) Toy truck (b) Airplane (c) Stool

(d) Bike (e) Fish

Figure 2. Medium (a-c) and small (d-e) images



(a) 427 CCPs (b) 533 CCPs (c) 203 CCPs



(d) 205 CCPs (e) 640 CCPs

Figure 3. CCP maps

4. Parallel search for partial paths

Various existing techniques learn their parameters from training images [8]. Because of the very large number of possible objects and images falling under the scope of our problem, proper formal training is hardly applicable here. It was found more appropriate to define the abstract category of multi-part objects using a limited number of local and global grouping criteria whose definitions were selected and validated on the basis of a new interactive methodology [11]. Details about the selected criteria appear below.

The GraphSearch deterministic algorithm [10] is used to build-up potential pieces of silhouettes. Due to the huge number of possible solutions, most nodes must be removed in order to keep the search under control. This is obtained by rejecting paths with intersections and by applying local grouping criteria. At the end of this procedure, a list of nodes containing paths of fifteen CCPs is produced. This number is typically not enough for the silhouettes of the interest object to be complete. This step is followed by the combination of nodes, as explained later. Scores are computed for the obtained final paths using global criteria.

4.1. Preprocessing

Short CCPs are removed from the initial pool of data. They are to be back in the final processing step, when small holes are filled in the combined nodes. Next, the number of elements is doubled by generating an oriented CCP for each orientation of remaining CCPs.

4.2. Local criteria

A new node must be validated by two tests in order to be added to the tree. The first and simplest is the distance between extreme points. The start point of the oriented CCP under test must be within a fixed distance of the last point of the current path. This threshold is currently set to 30 pixels. This distance check is only an optimization feature that allows to skip intersection checks if it is met. Further distance conditions will be applied later, together with other criteria. It is also to be noted that gaps larger than 30 pixels may be present in the final silhouettes obtained after combining partial paths.

The second test consists in intersection checks. Junctions are accepted only at extreme points. This condition eliminates a large number of possible paths. Gaps on paths are filled with lines before testing. Some valid nodes according to the above tests are not to be kept either. This other way to remove tree nodes is by the application of a set of local criteria to CCP paths. Failure to satisfy a set of conditions, in the form of a boolean equation, results in pruning of that node. Early removal of a node is more efficient. Hence, conditions are typically very restrictive in the early levels, and more permissive later on.

The ten local grouping criteria are listed in Table 1. Each has a simple formal definition. For instance, the two continuity criteria have a linear scale. A null angular difference between tangents at extreme points of the CCPs worths 100%. A 180 degrees difference worths zero. The arithmetic mean of scores on a path is computed. The distance criterion is the distance from end point of the last CCP of the path to the start point of the CCP to be added. The total length sums up the length of the gaps between consecutive CCPs and the length of the CCPs themselves. The number of parts criterion is computed using the number of concavities. It starts by filtering angles through the silhouette by grouping small angle variations together. Gaps on the boundary are filled. One is added to the result to take into consideration the main body of the multi-part object. Other criteria are also simple to compute.

Table 1. Local criteria encoding

Number	Criteria
1	Continuity
2	Unused
4	Distance
8	Number of parts
16	Surface area
32	Total length
64	Opening
128	Filled continuity
256	Obtuse angles
512	Hole proportion
1024	Early closure

Tree is generated up to level fifteen. After each level generation, the pruning process is executed. Figure 4 illustrates the principle. White nodes are the new nodes added and accepted by the GraphSearch. Local criteria are expressed in the form of a boolean equation, different for each level of the tree.

In the boolean equations, operands are pairs of numbers (criterion and its associated threshold) and operators are the AND (*), OR (+) and the ForceAND (X). The last one is used for optimization purpose and forces the respect of all conditions coming before it. Table 2 present the equations for the fifteen levels of the tree as applied to small images. Similar equations are used for medium images.

A common pattern is used all the way down the tree. It is continuity (#1) - distance (#4). It ensures a coherence between CCPs. Large distance asks for good continuity and smaller distances allow more discontinuities. Notice also the total length (#32) applied one level out of two in low levels, and more frequently at the end. This way, a chance

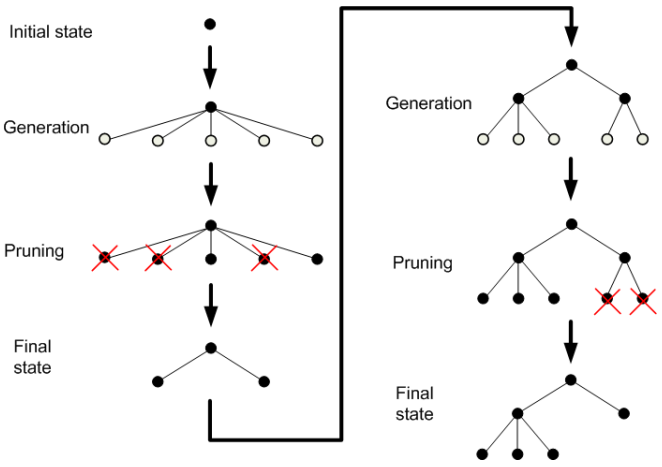


Figure 4. The generation-pruning switch

Table 2. Boolean equations for small images

Level	Associated equation
1	32 10
2	1 80 4 10 * 1 65 4 2 * + 4 1 +
3	1 80 4 10 * 1 65 4 4 * + 4 2 +
4	1 75 4 10 * 1 60 4 5 * + 4 2 + 512 10 * 32 50 * 1024 2 * 64 20 *
5	1 75 4 15 * 1 60 4 6 * + 512 5 + 512 12 * 128 55 * 256 2 * 1024 3 *
6	1 70 4 15 * 1 60 4 7 * + 512 5 + 128 55 * 8 1 8 2 + * 32 80 * 512 12 * 1024 4 *
7	1 70 4 15 * 1 60 4 8 * + 1 55 4 5 * + 256 3 * 1024 5 *
8	1 65 4 18 * 1 60 4 10 * + 128 65 * 32 100 * 512 15 * 1024 6 *
9	1 65 4 20 * 1 60 4 12 * + 128 75 * 512 15 * 32 120 * 1024 7 *
10	8 2 1 65 * 128 75 * 16 0.5 * 256 3 * 8 4 1 70 128 75 * 1 65 128 65 * 512 5 * + * 16 1 * 256 4 * + 512 15 * 4 20 * 32 150 * 1024 8 *
11 to 15	4 20 128 70 512 12 X

of survival is left to short paths. By level four, hole proportion, total length, early closure and opening criteria become meaningful. Early closure is used until the end because every newcomer must be tested for correct positioning. Level six is the first time the number of parts criterion is applied. Level ten groups are not self-sufficient. That is, they rarely represent the entire silhouette of the interest object. Thresholds are slackened as tree is expanded from level ten to level fifteen. The goal is to retrieve missing pieces so that when level fifteen groups are combined, it could result in a nearly complete silhouette.

From level ten, elimination process changes. All nodes are quite good because they meet tight conditions. The slackened criteria may not be sufficient to reduce combinatorial explosion. To help with that situation, a partial score is computed for each node using a linear combination of individual criteria scores. Then, only the best nodes are kept. This elimination process arises whenever the number of nodes at a given level exceeds 2000. It is then reduced down to 1500.

4.3. Global criteria and scores

Associating a quality score to a node is at the heart of the proposed method. A node is considered as either a completed or under-construction object. The former gets its value from the main score, the latter from the partial score.

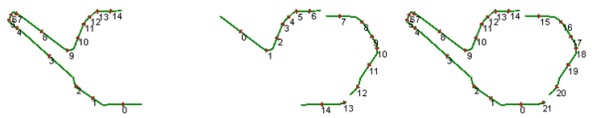
The main score has been mainly developed by Randrianarisoa et al. [11]. The ten global grouping criteria, with their weights, are as follows: closure (5), visual balance (1), compactness (1), number of parts (1), filled continuity (5), gap distribution (1), object-image position (1), surface area (5), border effect (1), hole proportion (1).

Only four elements are kept in the partial score, with unit weights: number of parts, filled continuity, surface area, hole proportion.

5. Combination of level 15 nodes

The combination procedure aims at producing complete silhouettes from partial paths. Only the best 500 best nodes are kept, according to partial score. 500 nodes generates 250000 pairs, which is quite enough in practice. For two paths to combine, they need not connect perfectly end point to start point. A partial combination is made of two paths that do not fit perfectly. A subset of primitives at the beginning of the second path is removed before adding it to the first path (see Figure 5). The minimum number of primitives to add to a node is set to 5 in experiments. Thus, the number of CCPs in combinations ranges from 20 to 30. Before accepting a tested combination, is must be validated by intersection checks and further application of local criteria.

In order to deal with both complete and incomplete objects, threshold values are permissive.



(a) Node 1 (b) Node 2 (c) Combination

Figure 5. Example of a simple combination

During execution, many node lists are kept up-to-date. One is called "End-answers". It contains potential final paths i.e. silhouettes. There are two conditions to be fulfilled in order to be in the list. First, direct distance between start and end point of path must be lower than or equal to 10% of the length of the path. Second, the line from start to end points of path must not generate intersections.

There are many objects with a silhouette of more than 30 CCPs. An optional step takes combinations of level fifteen and combines them again with level fifteen nodes. This way, solutions may contain from 25 to 45 CCPs, which is now enough for typical test images. This optional step is skipped when the best main score of the End-answers is at least as high as the best main score of the list of combined nodes and the number of CCPs in path of the best main score node of the End-answers is at least 15% of the number of CCPs after initial filtering.

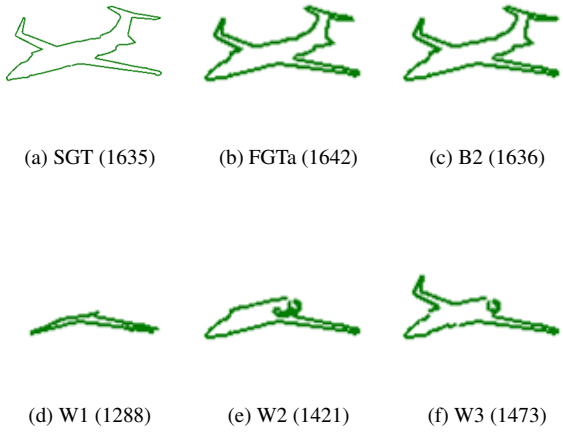
After combination and potential recombination, short CCPs initially removed are used to complete small gaps on the silhouette of the object.

6. Results

Results can be expressed in many ways. First of all, the best solution found (best main score solution, or FGTa) is compared to the human segmentation (SGT) and to other possible solutions (good or bad). This tells how hard it can be to discriminate good solutions from others. Next, the most similar solution to the SGT is located in the End-answers list. Then, the general usage of the CCP map throughout the search steps is shown and analyzed. Finally, evolution graphs display main score, similarity, precision, and recall results at various algorithmic steps.

6.1. SGT/FGT comparison

Figures 6-8 show results for three typical images. The main score of each solution appears in parentheses. The airplane is an interesting case in which one may easily find sub-objects, made of a subset of object parts. The image has background clutter and internal textures and markings. Many CCPs create bridges to go around some parts, like the plane's right wing. Pruning those CCPs helps converging to correct answers faster.



(a) SGT (1635) (b) FGTa (1642) (c) B2 (1636)

(d) W1 (1288) (e) W2 (1421) (f) W3 (1473)

Figure 6. Plane SGT/FGT: 2 best, 3 worst

The toy truck is another good case in which the method must face more than one good objects. Despite the numerous possibilities, the final FGTa is quite similar to the SGT reference. Remember that it is difficult to perfectly localize the SGT. In fact, its main score is 1663, compared to 1721 for the FGTa. That means, the method considers FGTa as a better multi-part object than SGT.

For the stool, many internal CCPs and holes enhance the complexity of this case by forming a lot of cycles and parallel routes. The junction points allow a contour to switch from one path to another, thus multiplying the number of possibilities to consider. Fortunately, the proposed method eliminates them rapidly.

6.2. Position of the most similar solution

In order to compare the obtained solutions with the SGT references, a similarity score is computed between silhouettes. This score considers common primitives on the paths and their length.

Table 3 show the position of the most similar solution computed by the method in the completed End-answers list. A value greater than 50% shows a notable similarity, but also with big differences. Greater than 70% means the differences will be minor. A similarity greater than 90% is obtained for solutions with almost invisible dissimilarities, or mostly meaningless differences. Position is an integer between one and the number of elements in the list. That total number of elements is in parentheses in the first column.

(a) SGT (1663) (b) FGTa (1721) (c) B2 (1697)

(d) W1 (1417) (e) W2 (1415) (f) W3 (1552)

(g) W4 (1570) (h) W5 (1605)

Figure 7. Toy truck SGT/FGT: 2 best, 5 worst

(a) SGT (1734) (b) FGTa (1741) (c) B2 (1738)

Figure 8. Stool SGT/FGT: 2 best

These numbers may be compared to the huge combinatorial number of possible solutions, as discussed earlier.

Table 3. Position of the most similar solutions

Image (#End-answers)	Position	Similarity
Juice (53)	3 (96%)	93%
Airplane (449)	2 (100%)	89%
Water can (2)	1 (100%)	100%
Angel fish (35)	3 (94%)	99%
Stool (427)	13 (97%)	92%
Toy truck (1277)	49 (96%)	87%
Fish (157)	1 (100%)	85%
Bike (857)	21 (98%)	90%
Hand (7)	3 (71%)	97%
Man (158)	1 (100%)	93%

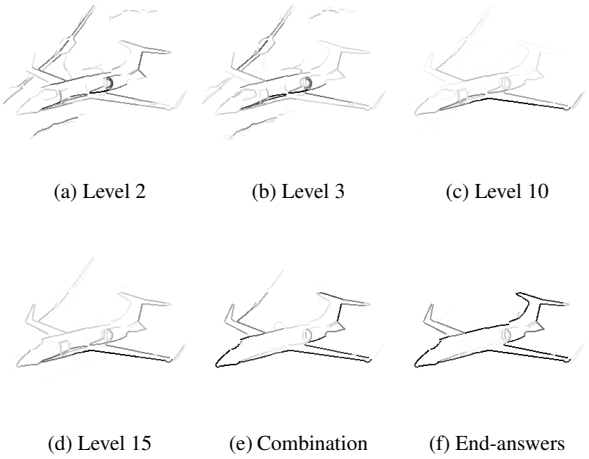
Only the toy truck, the bike, and the fish show notable differences between their most similar solution and their SGT. They are shown in figure 9.

(a) Fish (b) Toy truck (c) Bike

Figure 9. Most similar solutions to SGT

6.3. CCP-Usage

CCP-Usage tells how much each CCP is used in the generated solutions. In CCP-Usage maps, the darker the CCP, the more it is used at a given step. A normalized usage value is computed for each CCP at each algorithmic step, by considering the number of times it appears through the solutions of that step. CCP-Usage maps allow one to precisely and globally track when some background and texture features disappear or when some silhouette parts become stronger. Figure 10 shows CCP-Usage maps for the airplane. Background and texture noise rapidly disappear through levels of the tree. Only remains some spots, like the background road, due to its continuity and its very good connection with the airplane. Still, combinations successfully cut through this difficult case.



(a) Level 2 (b) Level 3 (c) Level 10

(d) Level 15 (e) Combination (f) End-answers

Figure 10. CCP-Usage maps

6.4. Evolution graphs

Four types of graphs are generated: the main score (Figure 11), the similarity (Figure 12), the precision (Figure 13(a)), and the recall (Figure 13(b)). Algorithmic steps 1 to 15 correspond to the tree levels, 16 is the first combination, 17 is the optional recombination, 18 is the cleaned End-answers list (according to similarity and main score), and 19 is the completed End-answers.

Precision is computed as the number of CCPs in the solution that are present in the SGT, divided by the total number of CCPs in the solution. It can be seen as the degree of purity of a solution. Recall is the number of CCPs in the solution that are present in the SGT, divided by the number of CCPs in the SGT. It computes the fraction of SGT CCPs successfully found.

At each step, a sample of one hundred solutions (the blue points in the graphs) are obtained by sampling the available solutions. For the main score, the solutions are obtained in the following way: a third from the best scores, a third from the middle ones, and the last third from the worst scores. For the precision and recall, the one hundred best solutions according to main score are retained. Finally, for the similarity with SGT, the one hundred best solutions according to similariy are selected. The red star in the graphs is the mean value of the selected solutions.

The range of the main score, from steps 1 to 15, is about 500 points for all images. To keep solutions as bad as 800 and as good as 1300 testify of the diversity of solutions in the tree. Combinations rapidly concentrate the scores to the ceiling. Similarity looks at the longest common path pieces. Its value grows linearly with tree level. The slope of this function is dependent on the number of CCPs in the SGT reference.

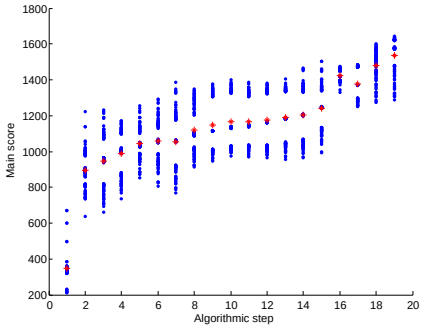


Figure 11. Main score graph for the airplane

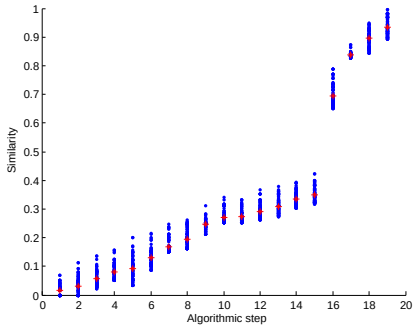
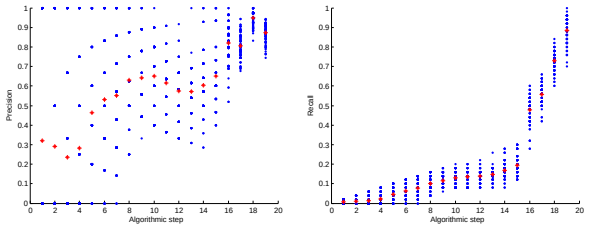


Figure 12. Similarity graph for the airplane

The precision behavior is intimately linked to the criteria used. In the first levels, precision takes only a few values due to the small number of possibilities. This creates graphs with a typically raising tendency from level 1 to 10 where severe constraints force the selection of CCPs on the boundary of the airplane. They are relaxed thereafter in order to find missing pieces of the silhouette. That permits "bad" CCPs to enter solutions, explaining the precision drop from levels 11 to 15. This is necessary in order not to miss good complete solutions. Recall also raises linearly during the tree generation. At each step, the recall score tops at the number of CCPs in a solution, divided by the number of CCPs in the SGT. So, the 100% mark is attained only at the end. Precision, however, can top 100% right at the beginning, and even drop after that, if the number of CCPs in SGT is low enough. For instance, an early good solution may be spoiled by adding bad CCPs.

7. Conclusions

A simple set of explicit local and global grouping criteria are combined to detect multi-part objects in complex images. A deterministic generic detection method based on parallel search tree expansion and pruning was developed and applied to a variety of noisy contour primitive maps. Input images show significant amounts of internal textures,



(a) Precision (b) Recall

Figure 13. Precision and recall graphs

markings, and background structure. The method is able to target the main subject of an image as long as it corresponds to a multi-part object of the proper complexity. It first builds small silhouette pieces made of fifteen oriented contour primitives, applying adapted local criteria at each tree level. These pieces are then combined and completed with small details using the remaining primitives. From the obtained object silhouette, it is straightforward to extract the corresponding region.

On a Pentium 4 2.0 GHz with 1024 MB of RAM, computation times range from 25 seconds for a simple image like the water can to 6.5 minutes for a complex image like the toy truck. The airplane requires about 4 minutes and the stool a little more. Algorithmic improvements may reside in criteria application, removing less useful ones, and also in the combination steps. Combination is repeated many times and a small timing improvement may provide important benefits. A parallel implementation is also likely to improve computing time. For instance, tree generation can be split by separating the first level nodes equally between processors. Each processor would make its nodes evolve until level 15 and communicate results to other processors for combination.

8. Acknowledgment

This work is supported by an NSERC discovery grant.

References

[1] J. Beis and D. Lowe. Learning indexing functions for 3-d model-based object recognition. In *Proc. AAAI Fall Symposium: Machine Learning in Computer Vision*, pages 275–280, 1993.

[2] G.-A. Bilodeau and R. Bergevin. Generic modeling of 3d objects from single 2d images. In *Proc. 15th International Conference on Pattern Recognition*, pages 770–773, 2000.

[3] G. Dorko and C. Schmid. Selection of scale-invariant parts for object recognition. In *Proc. of the 9th International Conference on Computer Vision (ICCV'03)*, pages 634–639, 2003.

[4] J. Elder and S. Zucker. Computing contour closure. In *Proc. 4th European Conference on Computer Vision*, pages 399–412, 1996.

[5] F. J. Estrada and A. D. Jepson. Perceptual grouping for contour extraction. In *Proc. 17th International Conference on Pattern Recognition*, pages 32–35, 2004.

[6] D. W. Jacobs. Robust and efficient detection of salient convex groups. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(1):23–37, January 1996.

[7] S. Mahamud, L. R. Williams, K. K. Thornber, and K. Xu. Segmentation of multiple salient closed contours from real images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(4):433–444, April 2003.

[8] D. R. Martin, C. C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(5):530–549, May 2004.

[9] M. Mokhtari and R. Bergevin. Generic multi-scale segmentation and curve approximation method. In *LNCS 2106: Scale-Space and Morphology in Computer Vision, Third International Conference*, pages 227–235, 2001.

[10] K. Nillson. *Principles of Artificial Intelligence*. Tioga, Palo Alto, CA, 1980.

[11] V. Randrianarisoa, J.-F. Bernier, and R. Bergevin. Detection of multi-part objects by top-down perceptual grouping. In *Proc. 2nd Canadian Conference on Computer and Robot Vision*, pages 536–543, 2005.

[12] D. Walther, U. Rutishauser, C. Koch, and P. Perona. Selective visual attention enables learning and recognition of multiple objects in cluttered scenes. *Computer Vision and Image Understanding*, pages 41–63, 2005.