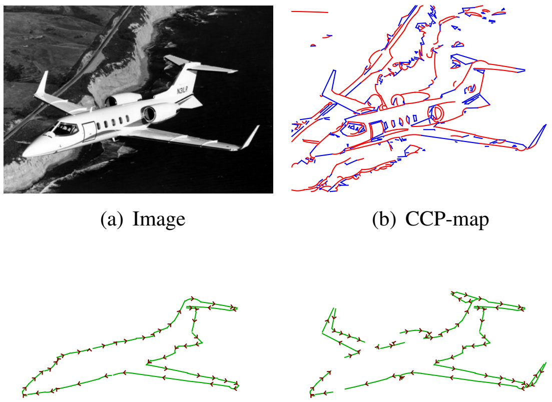
On structuring multiple grouping hypotheses in generic object detection

Guillaume Villeneuve and Robert Bergevin
Computer Vision and Systems Laboratory
Université Laval
Québec, QC, Canada
bergevin@gel.ulaval.ca

***Abstract*—A novel contour grouping method was recently proposed for the difficult task of detecting and delineating unexpected multi-part objects of unknown specific shape and appearance in a variety of natural images. This method, in many ways original and unique, was generally able to obtain object-level groups of quite good quality for a variety of objects and images. For each tested image, a number of object-level groups are hypothesized and ranked using a generic multi-criteria objective function. Experiments shown that object-level groups most similar to the human ground truth usually rank high. However, no object-level group was obtained with some difficult images. This paper proposes three important improvements to that original method. Firstly, fixed parameters are replaced by adaptive parameters, improving the robustness of the method even for the most difficult images. Secondly, a parallel version of the method is developed to either speed-up or scale-up the computation, making the method adaptive to practical space and time constraints. Finally, a further structuring of the object-level groups makes it possible to isolate the interesting ones and determine their number. A comparison with previous results illustrates the significance of the improvements.**

***Keywords*-cognitive computer vision, object detection, grouping, contour primitives, parallelization, clustering**


(a) Image

(b) CCP-map

(c) Best detection

(d) Ground truth

Figure 1. Best detected object by [1]. A CCP-map is a map of contour fragments of two generic types: straight-line segments (blue) and circular arcs (red). The best detected object is the candidate solution, an ordered subset of oriented CCPs from the CCP-map, with highest computed score. For a given image and associated CCP-map, the scores of the retained solutions are computed using a set of global shape criteria which is common to all object types. The ground truth is a manually-selected ordered subset of oriented CCPs from the CCP-map.

I. Introduction

Cognitive computer vision methods comprise many processing stages spanning different levels of description. In recent years, much emphasis has been properly laid upon making the descriptions at the object level more generic and categorical. A novel generic object detection (GOD) method was proposed in [1] based on a hierarchical contour grouping framework. It was shown that a strictly over-segmented feature map is a critical requirement best satisfied by constant-curvature contour primitives (CCP-map). Figure 1 presents the best unexpected object detected by that method, for a typical CCP-map, along with the human ground truth. Object-level groups are candidate solutions to GOD. Those candidate solutions are explored fully deterministically in [1] in contrast with, say, genetic algorithms where the candidate solutions are typically both initialized and explored randomly.

In [1], GOD was defined as delineating the objects of intermediate-complexity shape in static images, irrespective of their specific shape, color, texture, and imaging context. As such, it amounts to solving the figure-ground segmentation problem, a cognitive variant of the image segmentation problem, with the additional specification that the figure is an object of properly complex but unknown shape. A single method and parameter set is defined for any object whose structure, posture, and pose give rise to an intermediate-complexity shape in the image. In other words, and in contrast with most existing object detection methods in the literature, no category-specific feature-based model is ever defined and used. Instead, a generic set of shape criteria and score is relied upon in searching for the best candidate solutions.

As explained in [1], the rationale behind such a contrarian approach is very simple. In completely out-of-context situations where no specific object type is expected, it would not be practical to search for all possible types of objects in an image, either sequentially or simultaneously. The idea is to search instead for a more abstract category encompassing as many object types as possible in order to first detect any interesting, here defined as shape-salient, object.

A completely out-of-context situation is certainly an extreme situation. However, partially out-of-context situations are the norm for both humans and machines. In that respect, mainstream object detection methods take a quite extreme stance in that context is in turn restricted to a single object type which is searched for independently.

In [1], each grouping stage is formalized as a combinatorial multi-criteria optimization problem solved by systematically considering all possible groups. Using such an approach, a common problem arising at each processing level consists in ranking groups in order to identify the best ones. More generally, one needs to structure groups in order to isolate the interesting ones and determine their number.

This paper proposes using a clustering method to structure the object-level groups of [1] on the basis of their similarity and quality. It will be shown that, having adopted the hierarchical grouping framework, this structuring is obtained efficiently and effectively. Still, in order to benefit most from it, a sufficient number of groups of good quality must be obtained. Even though the set of groups was shown in [1] to be of good quality for a number of images of varying complexity, the most difficult images suffered from using fixed computational resources and parameters. A second contribution of this paper is a more efficient and robust computation of the groups.

II. Related work

Evidence has been published that perceptual grouping of contour primitives is key to image understanding [2]. However, few results are known regarding high-level grouping for GOD [3]. Contour grouping methods for object parts assume simple shapes [4]. As such, it becomes difficult to succesfully filter and structure the obtained groups towards GOD. A few recent generic contour extraction methods appropriately target unknown salient objects in natural images [5]. Closer to the approach in [1], those global optimization methods also detect and localize salient objects based on their shape in the image. Similarly to [1], such methods are generic as they do not make any assumption related to the object types. However, as discussed in [1], they typically choose to severely restrict the terms of their objective function in order to come up with a closed-form solution.

It was shown in [1] that such an approach is not robust in that it may easily produce a single best solution which is quite poor in terms of precision and recall of the ground truth features. With such an approach, in order to obtain a second-best closed-form solution, the problem set must be changed, e.g. by removing some image features. Typically, previous methods propose to remove all features from the best closed-form solution, which addresses the case of multiple salient objects in an image but not the problem of an erroneous best solution.

Instead, the method proposed in [1] restricts its multiple optimization criteria only in number but not in format, allowing higher-level criteria out of the scope of the closed-form methods. Besides, the number of retained candidate solutions is only limited by computational considerations, e.g. time and space limits. Retained solutions, which may overlap to any degree, are all ranked according to a single set of unrestricted criteria defining all intermediate-complexity shapes.

By explicitely considering and retaining partly-overlapping solutions (e.g. sharing a subset of contour primitives), the method in [1] offers a significant advantage over closed-form optimization methods. That is, alternative solutions may be structured and further analyzed, either automatically or interactively, in order to address more difficult contexts such as noisy feature maps, multiple interesting objects in an image, partly-occluded objects, etc.

In [1], the number of retained solutions is typically from a few to over one thousand. Properly adapting this number to the available computational resources and constraints was not attempted and would actually have been quite difficult given the number of parameters in the method. Besides, no further structuring of the retained solutions was attempted.

In this paper, we propose improvements to the method in [1] in order to better exploit its fundamental advantages. As a favorable experimental comparison with state-of-the-art global optimization methods was already performed in [1], this paper mostly presents and experimentally demonstrates improvements to [1] in terms of robustness, efficient adaptivity to computational resources, and structuring of the retained solutions. Specifically, it will be shown that ten or less significant solutions may be efficiently obtained for a superset of the test cases in [1].

A very interesting and related method for figure-ground segmentation was recently proposed in [6]. As in [1], hypotheses are made and ranked without prior knowledge of the properties of individual object classes or types. While sharing many interesting similarities with the approach introduced in [1], this new method differs with respect to a number of important design choices. Besides being based on region hypotheses instead of contours, an issue discussed in [1], the new method does not attempt to define the concept of interesting object as precisely as in [1]. For this reason, its scope and experimental evaluation methodology are different from [1], an object being defined mostly by its appearance in [6] whereas objects of interest in [1] are defined by their having an intermediate-complexity shape. Besides, results in [6] report only the best hypothesis for each ground-truth object, irrespective of its rank in the pool of region segments, whereas [1] compares very precisely the ordered set of contour segments in the best-ranked hypothesis with the corresponding ground-truth set.

Finally, a number of successful model-based object detection methods have also been proposed recently [7], [8]. Since they rely on matching partial or complete object contour templates, they are not properly addressing the GOD problem, defined as a figure-ground segmentation in [1] and [6]. Still, they need to address the structuring issue since possible matches are likely to overlap. However, that issue has yet to be given its due attention in model-based methods.

III. Object-level groups

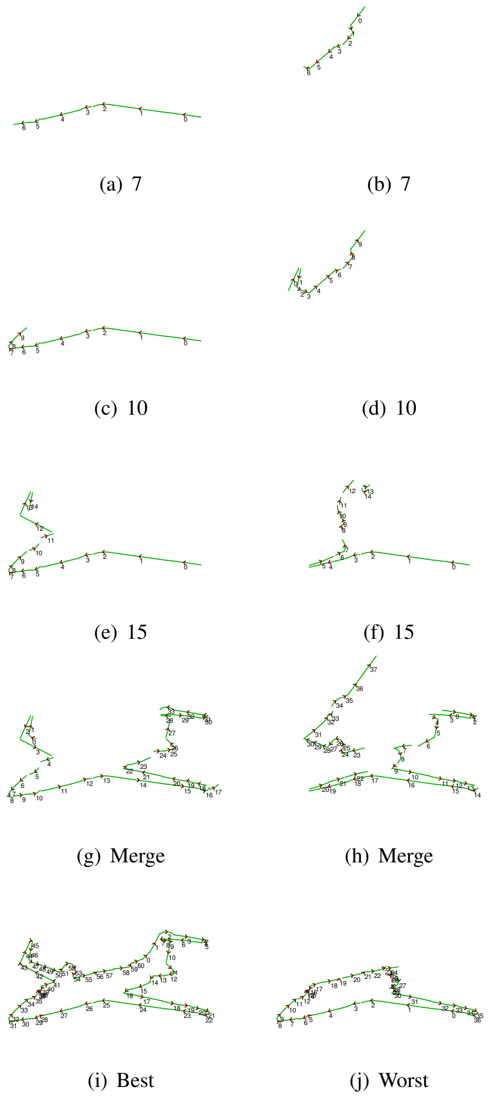
The original method in [1] generates multiple object-level grouping hypotheses from each input CCP-map, building 1-CCP groups and growing them gradually, one CCP at a time in the first 15 steps and by merging 15-CCP groups in the following two steps. At the nth step of the method, for n between 1 and 15, all possible n-CCP groups from the CCP-map are considered but only those satisfying a number of grouping constraints specific to that step are retained. The pruning constraints at each step are expressed as a specific boolean combination of a subset of ten intermediate-level contour grouping criteria with limiting parameters. In cases where too many groups are retained at a given step, the set is ranked and pruned to a fixed maximum size according to a partial score combining a subset of the intermediate-level contour grouping criteria.

The values for the limiting parameters in the boolean expressions are fixed and specific to each step and grouping criterion. Generic constraints are also added at each step in order to enforce non-intersecting boundaries and maximal-size gaps. Hence, the large number of parameters influencing both the quality and quantity of retained object-level groups. Despite this large parametric space, very good results were reported in [1] for a variety of images and objects.

It is to be noted that a n-CCP group is actually an ordered (numbers in Figure 2) list of oriented CCPs, hence one of two possible directions (arrows in Figure 2) is assigned to each CCP in the group. In practice, the number of possible groups is exponential and most of them need to be removed at each step. After 15 steps, a limited number of 15-CCP groups are obtained.

Usually, 15-CCP groups cover at best only a part of the object boundary. One or two attempts at merging pairs of partly-overlapping 15-CCP groups follow. Finally, a single instance is kept for each set of mostly identical groups and it is completed using small still-unused CCPs from the input map. Overall, there is a total of 19 steps needed to typically retain from a only a few up to over a thousand final groups from an exponentionally-large set of possible groups.

As mentionned earlier, the obtained object-level groups are ranked automatically using a multi-criteria main scoring function. In [1], the method was shown to produce high-ranking high-quality object-level groups for diverse multi-part objects. Figure 2 shows two example boundaries (one totally on the object and one partly in the background) obtained using the CCP-map in Figure 1(b) after grouping steps number 7, 10, 15, and 17 (second merge), along with the best and the worst retained completed object-level groups.


(a) 7

(b) 7

(c) 10

(d) 10

(e) 15

(f) 15

(g) Merge

(h) Merge

(i) Best

(j) Worst

Figure 2. CCP groups after different grouping steps. Groups (a), (c), (e), and (g) are on the object boundary. Groups (b), (d), (f), and (h) are under-segmented, partly overlapping the background. Group (i) is the top-ranked retained final group. Group (j) is the bottom-ranked retained final group. Up to a few thousands groups may be retained at each step.

A. Robustness

Having a large number of fixed parameters is problematic for at least two reasons. Firstly, it makes it difficult to properly adjust the parameters in order to obtain the best possible results within given space and time limits. Secondly, fixed parameters may result in no group being retained at a given grouping step for the most difficult images. In such a case, no object-level group is obtained by the method, as reported in [1].

In order to extend the applicability of the method, one could attempt to change the constraints or their parameters in the first 15 steps. Given the scope of the method and the dimensionality of the problem, even a machine learning approach to parameter selection appears difficult. Another possibily could be to automatically adapt the parameters to each single test case, making sure the retained groups at each step are obtained within acceptable time limits, their number being within acceptable space limits. Again, it appears difficult to obtain such a result with a deterministic method. A different and much simpler adaptive approach is proposed here in which the constraints are actually removed.

More precisely, the partial score optionally used to limit the number of retained groups at each step is now computed as each possible group is considered in steps 2-10 in order to always keep a limited-size list of the best groups. The minimum CCP length in step 1 is now made relative to the image dimensions. Intersection and neighborhood generic constraints are kept at each step, the latter being generalized to either adapt to image dimensions or ensure a minimum number of neighbors, irrespective of the gap size. For steps 11-15, an iterative pruning scheme is adopted based on ranking and pruning the groups according, in turn, to each of the four criteria used as constraints in [1]. A fixed number of groups are kept at each step. A similar but simpler approach is taken for the two merging steps in which fixed thresholds are replaced by an integrated score.

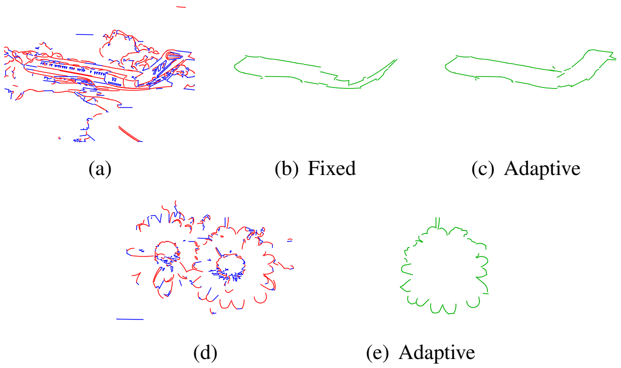
Overall, those changes to the method in [1] makes it adaptive to the image size and content, ensuring a comparable number of object-level groups for any input CCP-map. Besides, as explained in the following subsection, it makes it possible to adapt the exploration of groups to the available computational resources.

Figure 3 shows the best object-level group obtained with the new adaptive method on two CCP-maps. For the first CCP-map, the adaptive method produces a visually better result. For the second CCP-map, the obtained result is quite good whereas the method described in [1] did not find any object-level group for that map.

Figure 4 presents additional results for CCP-maps obtained from [1] and computed from ETHZ and BSD standard image datasets. The first two results using CCP-maps from [1] are very good, the plane showing a significant improvement over the result of the original method (Figure 1(c)). The four ETHZ swan results compare favorably with state-of-the-art model-based detections. The last two results from BSD are comparable to state-of-the-art contour extraction [5], despite using CCP-maps built from a simpler Canny edge detector.

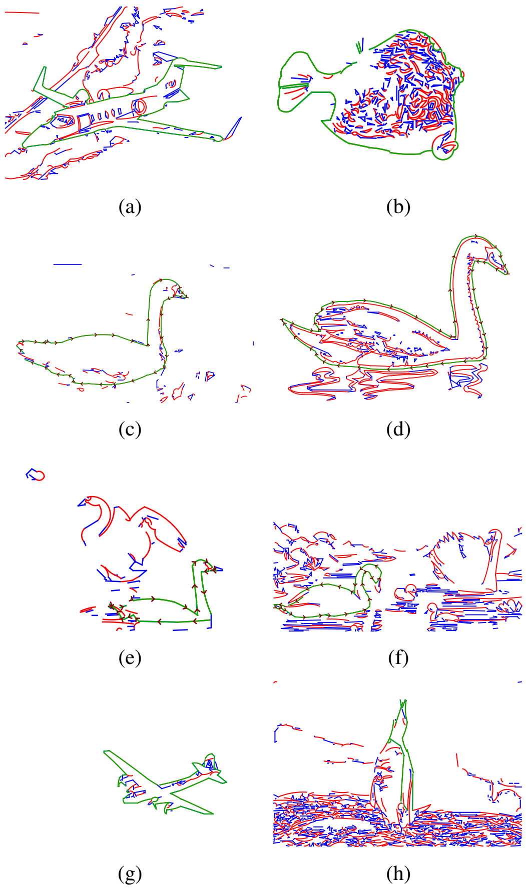
B. Efficiency

The improved method just described offers many opportunities for parallellization. For instance, 1-CCP groups may be evenly divided into M subsets, each one computing steps 2-15 independently for its groups. Another finer-grained parallelization may instead be applied at each step. For steps 2-10, each group expansion may be evaluated independently.


(a) (b) Fixed (c) Adaptive

(d) (e) Adaptive

Figure 3. Improved best group. CCP-map (a) is a difficult one since it contains a large number of distracting CCPs. The adaptive method produces an improved result. CCP-map (d) was actually too difficult for the original method. The adaptive method produces a visually good result.


(a) (b)

(c) (d)

(e) (f)

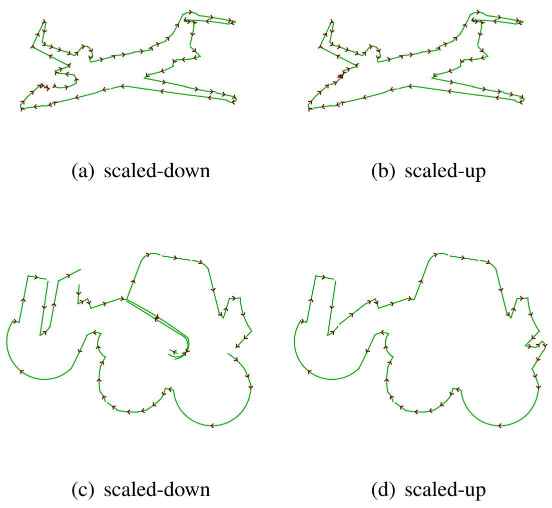
(g) (h)

Figure 4. Additional results. The adaptive method produces visually good results, comparable to or better than state-of-the-art methods.

For steps 11-15, each of the four parameters needs to be evaluated sequentially but, for each evaluation, each group may be evaluated independently. It was empirically determined that division into subsets for steps 2-10 and fine-grained parallelization for steps 11-15 produce good results, as reported below. In steps 16 and 17, possible pairs of 15-CCP groups are divided into M subsets evaluated in parallel.

Using a parallel method, one may obtain either the same groups faster, keeping all parameters and computations unchanged, or better groups within the same time limits, exploring a larger set of possible groups. Both speed-up and scale-up may be useful in practice.

Using the first ten CCP-maps from [1] (reproduced in Figure 5), the average speed-up obtained is more that 2 when run on a quad-core PC (an average computation time of about 30 seconds per image). The speed-up is with respect to a new optimized sequential implementation of the method in [1] in which pairwise distances between CCP extremities are pre-computed in parallel. Figure 6 shows that scaling the exploration of groups up or down changes the quality of the best object-level group for images of typical complexity. Quantitatively, it was observed that the F-measure varies proportionally to the scaling.


(a) scaled-down

(b) scaled-up

(c) scaled-down

(d) scaled-up

Figure 6. Adaptation to available computational resources. With the new adaptive method, scaling the computation up or down is easy. The quality of the best object-level group improves as the number of retained groups is increased at each step. This number is increased (reduced) as the computation is scaled up (down).

IV. Structuring method

After the final step in [1], the number of groups may range from only a few to over a thousand. However, many hypotheses may be similar and, as explained in [6], filtering redundant hypotheses is important in order to diversify the best-ranked retained hypotheses. It is proposed here to structure the object-level groups using a hierarchical clustering method. A nearest-neighbor rule is used and a simple pairwise similarity is computed on the ordered set of oriented CCPs forming each group. A prototype group is then selected from each cluster.

A. Pairwise similarity

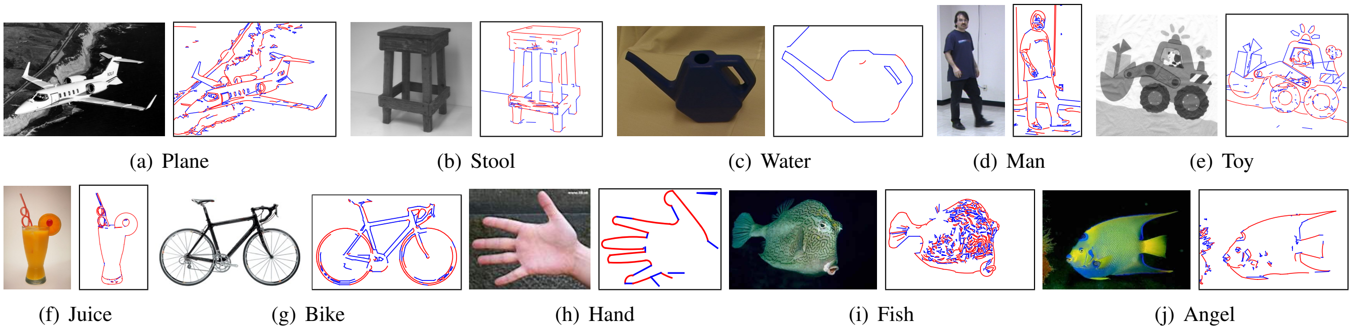
Each object-level group is made up of an ordered list of oriented CCPs. Two groups made up of the same oriented CCPs in the same (or reverse) order are maximally similar. Non-parametric statistical measures such as Spearman's rank correlation coefficient allow comparison of ordered lists of the same length but they do not consider binary parameters, e.g. orientation, for the list items. Actually, tens of similarity measures have been proposed for ranked lists of the same items [9]. On the other hand, recent specialized methods e.g. for computing similarity of ordered gene lists are complex and domain-specific.

Here, the two groups to be compared are cyclic ranked list in that the starting CCP is arbitrary on the closed contour. Besides, a missing or spurious CCP in the second group, a change of position, or a change of orientation should decrease the pairwise similarity. Given these specific requirements, a simple and efficient definition of the similarity of a pair of groups a and b is adopted. It corresponds to the proportion, relative to the size of the largest group, of the size of the largest subset of consecutive CCPs in one group that appear in the same order and orientation in the second group. Given that only the largest properly ordered and oriented overlapping subgroup is considered, the obtained similarities are likely to decrease rapidly as the two groups start differing.

B. Hierarchical clustering

Clusters are obtained at leaf nodes of a divisive binary tree using algorithm 1. Starting with a root node containing all object-level groups, each node is partitioned in two descending nodes until all pairwise similarities of groups within it are over a fixed threshold. The two groups with the minimum pairwise similarity become the representatives of the new nodes and the other groups are assigned to the node with the highest similarity with its representative.

Figure 7 presents normalized CCP maps linearly combining the object-level groups at each node of the clustering tree. The minimum pairwise similarity within a leaf cluster was set to $\theta = 0.6$. The smaller that value, the smaller the size of the clustering tree and the smaller the number of leaf nodes and prototype groups. In a normalized CCP map, each CCP is displayed using a grey level proportional to the number of times it appears in different maps, here corresponding to the different groups in a cluster at a given node. Leaf clusters have more uniform normalized CCP-maps than non-leaf clusters. Normalized CCP-maps of leaf clusters are of diverse shapes. Interestingly, groups in each


(a) Plane (b) Stool (c) Water (d) Man (e) Toy

(f) Juice (g) Bike (h) Hand (i) Fish (j) Angel

Figure 5. Images and CCP-maps reproduced from [1].

Algorithm 1 Hierarchical clustering

```
Initialize root node N with all groups
$OPEN \Leftarrow \{N\}$
while $OPEN$ is not empty do
    $C \Leftarrow OPEN$.first
    $OPEN$.pull_front
    $ms \Leftarrow$ minimum pairwise similarity in C
    if $ms \geq \theta$ then
        continue while
    end if
    $a, b \Leftarrow$ groups with minimum pairwise similarity in C
    $N1 \Leftarrow$ groups closer to a
    $N2 \Leftarrow$ groups closer to b
    $OPEN \Leftarrow \{ OPEN\ N1\ N2 \}$
end while
```

leaf cluster were observed to be also structured according to their main score ranks and their similarity with ground truth.

Figure 8 presents all object-level groups in the leaf cluster at node 4. Groups are in descending order according to main score. Clustered groups have similar shapes, as expected.

C. Prototype selection

Apart from its proposed use in structuring multiple grouping hypotheses, clustering is also a popular framework to organize training instances in k-nearest-neighbor (kNN) classification. In such a case, selection of a good prototype for each cluster is an important issue [10]. In order to be robust to possible outliers, it is proposed to define the prototype as the group with best main score, irrespective of the pairwise similarities. In this way, prototype selection is made specific to the problem at hand, while still benefiting from the generic structuring of groups into clusters. A similar choice was made in [6].

Figure 9 presents the prototypes in different leaf clusters sorted with the best ones first. Nodes are numbered as in Figure 7. Prototypes are of diverse intermediate-complexity shapes, as expected.

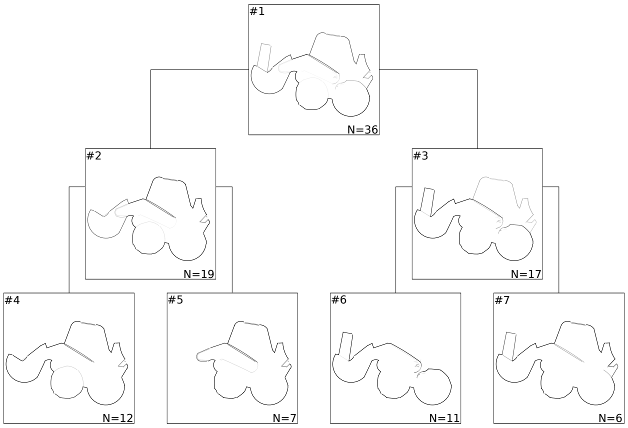
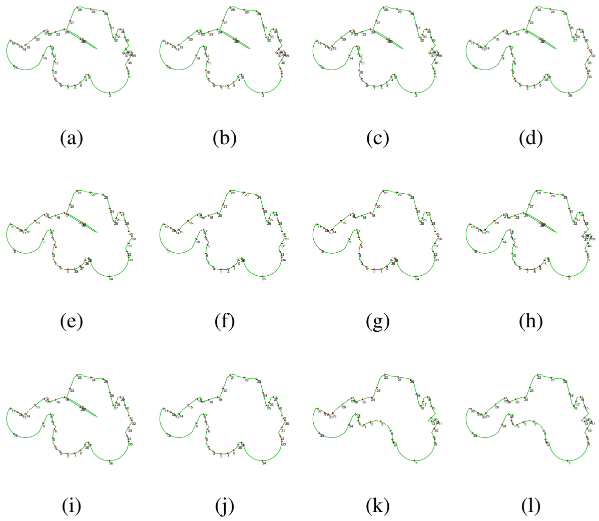


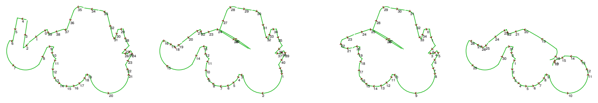
Figure 7. Tree obtained by hierarchical clustering using CCP-map in Figure 5(e). Nodes are represented by normalized CCP-maps. Nodes are numbered from 1 according to a breadth-first traversal of the binary tree. One may notice the diversity of shapes in the leaf clusters.


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

(e) (f) (g) (h)

(i) (j) (k) (l)

Figure 8. Leaf cluster groups. All object-level groups in leaf cluster #4 are displayed. One may notice the similarity of shapes in the leaf cluster.



(a) 7 (b) 4 (c) 5 (d) 6

Figure 9. Prototypes of leaf clusters. Prototypes are ordered with decreasing main score. One may notice the similarity of prototypes with the normalized maps of leaf clusters.

V. Results

Using the same ten CCP-maps as before (Figure 5), the average precision and recall of the best object-level groups are 80% and 83%, respectively, being absolutely improved by 1% and 6%, respectively, with respect to the original method in [1]. The scaling of the adaptive method was chosen in order to keep the average computation time unchanged. The worst recall (for CCP-map (i) in Figure 5) went from 39% to 67%, for an 8% decrease in precision. Table I(a) shows quantitative results for each image. Importantly, precision and recall is very strict here, being computed using CCP labels i.e. imposing perfect correspondance and ordering of contour primitives. In contrast, a benchmark evaluation standard based on partial overlap of enclosed image regions or bounding-boxes is used in the VOC Challenge and in [6]. As an indication, the computed precision and recall of the object-level group in Figure 1(c) are 81% and 61%, respectively. The improvement on images that already produced quite good results validates that the added robustness, efficiency, and flexibility of the new method is not obtained at the expense of poorer results.

In another experiment, the maximum number of retained groups and the neighborhood extent were adapted to the available computation time. In practice, a speed-up factor is chosen according to the available scale-up i.e. architecture. For the ten CCP-maps, the average absolute difference in F-measure for a speed-up factor of 1/17 (computation is 17 times longer) is 2%. Conversely, for a speed-up factor of 7 (computation is 7 times shorter), the difference is -5%.

The mentioned ten images were selected in [1] as representative of major difficulties faced by object detection methods, which was made clear in the experimental evaluation section of [1]. Other experiments were made on two additional datasets, each of ten images. In both cases, the parameters optimized for the first ten images were left unchanged. The first additional dataset (quantitative results shown in Table I(b)) gave a mean difference of -1% and 4% in precision and recall, respectively as compared with [1], showing the generalization capability of the adaptive method. The second additional dataset comprised ten difficult images that produced no results at all in [1]; good to excellent detections were obtained with half of them.

Detailed results in Figure 10 for a selection of images from those three datasets demonstrate the scope of the method and the quality of its results. One may notice the effectiveness of the clustering and prototype selection steps.

VI. Conclusion

Three significant improvements were proposed to a recent generic object detection method, making it adaptive to space and time limits, while being more robust and efficient. A comparison with previous results illustrated the new capabilities. Future experiments are planned to assess the possible benefits of a highly scaled-up computation on a high-performance cluster. It is expected that more complex contexts could then be in the scope of the method, such as partly-occluded objects, multiple objects, and even moving objects in image sequences.

Acknowledgment

This work is financially supported by a Natural Sciences and Engineering Research Council of Canada (NSERC) discovery grant.

References

[1] R. Bergevin and J.-F. Bernier, "Detection of unexpected multi-part objects from segmented contour maps," *Pattern Recognition*, vol. 42, no. 11, pp. 2403–2420, 2009.

[2] J. Elder and S. Zucker, "Evidence for boundary-specific grouping," *Vision Research*, vol. 38, pp. 143–152, 1998.

[3] J. Feldman, "The role of objects in perceptual grouping," *Acta Psychologica*, vol. 102, pp. 137–163, 1999.

[4] P. Sala and S. J. Dickinson, "Contour grouping and abstraction using simple part models," in *ECCV (5)*, 2010, pp. 603–616.

[5] F. Estrada and J. Elder, "Multi-scale contour extraction based on natural image statistics," in *Proc. IEEE Workshop on Perceptual Organization in Computer Vision*, 2006.

[6] J. Carreira and C. Sminchisescu, "Cpmc: Automatic object segmentation using constrained parametric min-cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1312–1328, 2012.

[7] X. Yang, H. Liu, and L. J. Latecki, "Contour-based object detection as dominant set computation," *Pattern Recognition*, vol. 45, no. 5, pp. 1927–1936, 2012.

[8] Q. Zhu, L. Wang, Y. Wu, and J. Shi, "Contour context selection for object detection: A set-to-set contour matching approach," in *ECCV (2)*, 2008, pp. 774–787.

[9] W. Webber, A. Moffat, and J. Zobel, "A similarity measure for indefinite rankings," *ACM Trans. Inf. Syst.*, vol. 28, no. 4, pp. 20:1–20:38, Nov. 2010. [Online]. Available: http://doi.acm.org/10.1145/1852102.1852106

[10] S. García, J. Derrac, J. R. Cano, and F. Herrera, "Prototype selection for nearest neighbor classification: Taxonomy and empirical study," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 3, pp. 417–435, 2012.

Table I
COMPARISON OF ADAPTIVE AND FIXED (ORIGINAL) METHODS. DATASET 1 IS USED BY THE ADAPTIVE METHOD TO OPTIMIZE ITS PARAMETERS. DATASET 2 KEEPS THE PARAMETERS UNCHANGED.

(a) Dataset 1

Image	Precision		Recall		Similarity	
—	Ad.	[1]	Ad.	[1]	Ad.	[1]
Plane	0.738	0.809	0.726	0.613	0.823	0.691
Stool	0.900	0.903	0.973	1.000	0.967	0.947
Water	0.958	0.960	0.958	1.000	0.983	0.938
Man	0.818	0.824	0.750	0.778	0.795	0.795
Toy	0.900	0.619	0.735	0.531	0.796	0.606
Juice	0.688	0.535	0.880	0.920	0.854	0.794
Bike	0.936	0.936	0.746	0.746	0.865	0.857
Hand	0.852	0.857	0.958	1.000	0.921	0.866
Fish	0.527	0.607	0.674	0.395	0.714	0.534
Angel	0.632	0.769	0.923	0.769	0.847	0.767
Average	0.795	0.782	0.832	0.775	0.856	0.779

(b) Dataset 2

Image	Precision		Recall		Similarity	
—	Ad.	[1]	Ad.	[1]	Ad.	[1]
Plane03	0,550	0,457	0,647	0,471	0,635	0,490
Plane13	0,556	0,586	0,541	0,459	0,662	0,531
Plane21	0,893	0,880	0,862	0,759	0,837	0,741
Compass	0,826	0,760	0,633	0,633	0,789	0,808
Stool 2	0,781	0,697	0,833	0,767	0,791	0,674
Lamp	0,897	0,931	0,839	0,871	0,858	0,833
Chair	0,893	0,720	0,714	0,514	0,804	0,537
Hat	0,391	0,625	0,529	0,588	0,758	0,729
Angel 2	0,727	0,739	0,865	0,919	0,890	0,859
Cup	0,481	0,684	0,929	0,929	0,786	0,893
Average	0,700	0,708	0,739	0,691	0,781	0,710

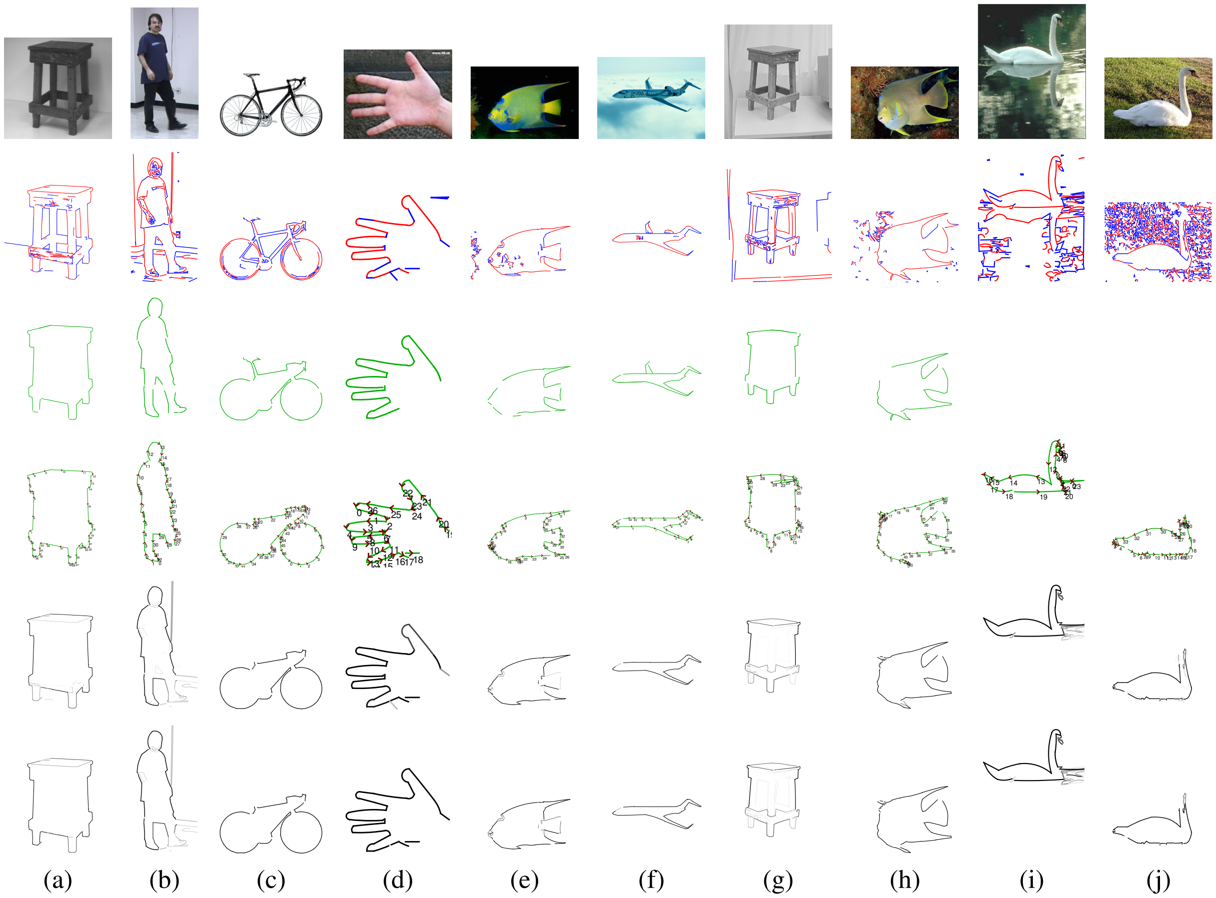


Figure 10. Detailed results. Each column presents, from top to bottom: the input image, the computed CCP-Map, the object-level ground-truth group, the best object-level group detected, a normalized map of all retained object-level groups, and a normalized map of object-level prototype groups selected from the leaf clusters. Images (a) - (e) are from the dataset used in [1] (see Figure 5), images (f) - (h) are from the first additional dataset (see Table I(b)), and images (i) - (j) are from the ETHZ dataset (with ground truth left undefined).