Click'n'Cut: Crowdsourced Interactive Segmentation with Object Candidates

Axel Carlier
Vincent Charvillat
IRIT-ENSEEIHT
University of Toulouse
Toulouse, France
axel.carlier@enseeiht.fr

Amaia Salvador
Xavier Giro-i-Nieto
Universitat Politecnica de Catalunya
Barcelona, Catalonia
xavier.giro@upc.edu

Oge Marques
Florida Atlantic University
Boca Raton, Florida (USA)
omarques@fau.edu

ABSTRACT

This paper introduces *Click'n'Cut*, a novel web tool for interactive object segmentation designed for crowdsourcing tasks. *Click'n'Cut* combines bounding boxes and clicks generated by workers to obtain accurate object segmentations. These segmentations are created by combining precomputed object candidates in a light computational fashion that allows an immediate response from the interface. *Click'n'Cut* has been tested with a crowdsourcing campaign to annotate images from publicly available datasets. Results are competitive with state-of-the-art approaches, especially in terms of time needed to converge to a high quality segmentation.

Categories and Subject Descriptors

H.1.2 [**Models and Principles**]: User/machine Systems, Human factors; I.4.6 [**Image Processing and Computer VisionI**]: Segmentation, Pixel classification

General Terms

Algorithms, Design, Experimentation, Human Factors

Keywords

Crowdsourcing; object segmentation; object candidates

1. MOTIVATION

Computer vision is an active research area, which has became increasingly relevant and pervasive, thanks to the growing amount of ubiquitous cameras that generate large amounts of visual data. A classic problem in computer vision systems is object detection, a task in which machines are expected to locate and recognize the objects present in a scene with an accuracy similar or better than a human would achieve. The mapping between the quantized pixels analyzed by computers and the concepts with which the human mind seems to operate has been referred as the *semantic gap*. Despite continuous advances in the field, such a gap has not been completely bridged yet.

CrowdMM'14, November 7, 2014, Orlando, FL, USA.

http://dx.doi.org/10.1145/2660114.2660125 .

This paper addresses the problem of how to leverage the human power of the crowd into solving the problem of pixel-wise object segmentation. The goal of an object segmentation process is to label all pixels of an image depending on whether they belong to a certain object or not. Typically, the output of such process is a binary mask whose white pixels represent the object and the black pixels correspond to the background. Our work explores how crowdsourcing can be used as a valid strategy for obtaining a large amount of high quality object segmentation results (masks), as long as the appropriate tools and data collection strategies are adopted.

The main contribution of this paper is the design and test of an online web interface (*Click'n'Cut*) that is responsive to the users' interactions and guides them into collecting highly informative data. Our results indicate that this tool is competitive with other state-of-the-art interactive segmentation techniques, especially in terms of responsiveness. Additionally, we present a technique to filter out noisy user interactions, specifically designed for the object segmentation task.

2. RELATED WORK

Interactive Object Segmentation. The segmentation of objects by combining human interaction and image processing algorithms has been extensively explored in the literature. In such interactive setup, the graphical user interface responds to some sort of weak annotation (e.g., bounding box, scribbles, or clicks) from the user by generating and displaying a complete segmentation of the object. The typical workflow expects the user to interact with the proposed solution either by accepting it or by providing more traces that may allow the segmentation algorithm to converge to a satisfactory result. Most interactive segmentation techniques normally propagate the user-generated labels of *foreground* and *background* pixels through a graph-based representation of the image.

The foundational proposal for interactive foreground segmentation was based on *graph cuts* [5, 16]. The algorithm considers every pixel as a node in a graph, connected to their spatial neighbors by an edge whose weight depends on the visual similarity between pixels. Segmenting an object is equivalent to minimizing an energy function defined over the graph. *Partition-based solutions* [15, 13] avoid the computational load of a pixel-by-pixel segmentation by working with unsupervised image segmentations performed offline. The adjacency information between regions coded in a partition can be further enriched by iteratively merging pairs

of neighboring regions and defining a *hierarchical partition*. Interactive segmentation solutions such as [18, 1, 2, 9] use these data structures to propagate labels through different spatial scales. The comparative study in [14] indicated similar accuracy labels for GrabCut [16] and hierarchical solutions [18, 1], but a faster response for the latter ones.

The solution adopted in our work does not solve any labeling of a graph; instead, it generates segmentations by combining a precomputed set of object candidates (also referred to as "saliency detectors") [6, 3]. These techniques basically train a classifier to estimate the "objectness" of a pool of regions. Our approach is inspired by [19], where crowdsourced clicks labelled as *foreground* or *background* were mapped into a collection of object candidates to select the region which better matched the captured traces. However, our system is more flexible than that, because it obtains solutions that can combine multiple candidates.

Crowd-based Object Annotation. Most initiatives for object annotations from the crowds have adopted a *collaborative* approach, where users are instructed on how to generate high quality segmentations. Incentives for the workers may vary from an abstract call to help science, to a very accurate pricing policy.

A popular initiative in this direction is *LabelMe* [17], an online platform that has collected a large amount of local annotations by asking volunteers to draw a polygon around the object. One of the most ambitious projects up to date is related to the Microsoft COCO (Common Objects in COntext) dataset [11]; this segmentation effort uses the *OpenSurfaces* interface [4], an open source tool based on polygonal segmentation. The crowd was also used in [10] to assess an interface aimed at choosing the best input modality among a bounding box, a sloppy contour or a tight polygon. The authors highlight that in crowdsourced campaigns the *annotation time* is the basic budget constraint, and that by automatically adapting the annotation mode to the image it is possible to optimize the quantity and quality of the segmentations. Table 1 summarizes the characteristic figures of these related works.

In our work we have tried to adjust as much as possible to the experiment described in [14] to be able to compare the quality of an online crowdsourced solution with respect to an offline campaign with expert annotators.

3. CLICK'N'CUT

In this section we describe our web interface for interactive object segmentation, *Click'n'Cut* (Figure 1). It displays the image that we wish to segment, along with a set of basic interactions (on the bottom-right of the screen) and a reminder of how the interface works (on the top-right part of the screen). There is also a description of the object to segment on the top of the screen, right above the image.

The fundamental interactions available to the worker are the left and right clicks. A left click on the image indicates a *foreground* point (in green) whereas a right click on the image indicates a *background* point (in red). After each click the current version of the segmentation is updated and displayed over the image with an alpha value of 0.5 by default. At any time the worker can choose to modify the alpha using the *Transparency* slider to either get a better look at the image or to better see the current foreground mask.

A worker can also correct a wrong click by just clicking on it again to make it disappear. The *Clear points* button

Figure 1: Screenshot of the Click'n'Cut interface.

removes the entire set of clicks that have been made by the worker. Finally, once satisfied with the result, the worker can go on to the next task by clicking the *Done* button.

The interface is implemented using HTML5 and JavaScript on the client side, and Java on the server side. The server side handles the computation of the current best mask as well as the persistence of workers' interactions in the database.

To compute the best mask with respect to a set of f *foreground* points and b *background* points, we adopt the following algorithm.

For each mask $m \in MCG$, where MCG is the set of masks computed using the *Multiscale Combinatioral Grouping* algorithm [3] , we start by computing two scores fg_m and bg_m. fg_m (resp. bg_m) is the number of foreground (resp. background) points that are correct with respect to m. If there exists a mask m^* for which $fg_{m^*} = f$ and $bg_{m^*} = b$ then m^* is the best possible mask and this is the mask that will be shown to the worker. Else, it means that no mask is correct with respect to all worker's clicks. In that case, we build a set of masks $M^* = \{m \in MCG, bg_m = b \text{ and } fg_m > 0\}$. This means that M^* contains the masks that have not been defined as background and for which there is at least one foreground point. The union of all masks that belong to M^* generate the mask that is displayed to the worker.

4. EXPERIMENTS AND RESULTS

The experimental setup used for evaluation has been proposed by [14]: it consists of 100 objects to segment from 96 different images from the Berkeley Segmentation DataSet (BSDS) [12]. Each object is described by a sentence in English and a binary mask is provided as ground truth. We augmented the dataset with 5 more images (from the PASCAL VOC dataset [8]) in order to introduce gold standard tasks that would allow quality assessment of the workers' performance while working on those tasks.

4.1 Protocol

We have structured our experiments into two campaigns: (i) we ask workers to use the *Click'n'Cut* interface to segment the objects; and (ii) we ask experts to perform the same task.

4.1.1 Campaign #1: segmenting objects with a crowd

We started a campaign on `microworkers.com`, for which we created 20 jobs that consisted in segmenting 105 objects. We paid 4 USD for the job, which means that each segmented object is worth 3.8 cents. We also added an extra incentive in the middle of the campaign: we told workers that the top three performers would receive an extra bonus of 5 USD. A total of 99 users participated to the campaign but only 20 workers completed all the 105 tasks.

4.1.2 Campaign #2: segmenting objects with experts

We conducted the exact same study from the first campaign except that we asked experts from different vision labs to interactively segment the objects using *Click'n'Cut*. We did not pay the experts; we simply asked them to try to reach for the best possible segmentation results. A total of 15 experts (11 males, 4 females) participated in this study, with ages ranging from 19 to 55.

4.2 Results

4.2.1 Quality of the traces

Our first study focused on the quality of the collected traces. Crowdsourced traces are usually noisy due to several reasons, chief among them the misunderstanding of the instructions for the task, e.g., selecting a larger/different region than the requested one.

The quality of the users was estimated using five gold images from the Pascal VOC dataset [8]. The comparison between the error percentage for the gold or test datasets (Figure 2) indicates that seven users (labeled 3, 5, 8, 10, 11, 18 and 19 in Figure 2) were successfully identified as unreliable because they performed worse than a 20% error threshold on both the gold and test images. It also shows that workers 12 and 15 performed significantly worse on the gold images than on the test images. The high error rates from users 3 and 19 were due to their opposite interpretation of the foreground/background clicks and masks.

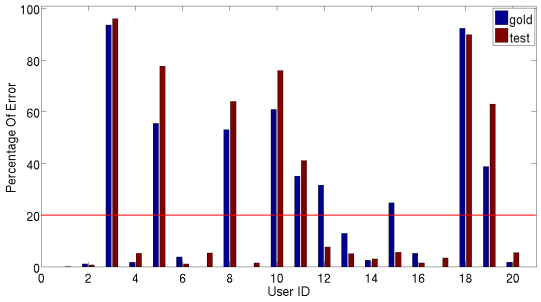

Figure 2: Error rate per worker for the test and the gold datasets.

4.2.2 Accuracy vs. time trade-off

Our interactive segmentation tool was compared with the top two best configurations analysed in [14]: GrabCut [16] and hierarchical partition with BPTs (BPT) [1, 18]. The different solutions are assessed in terms of the accuracy vs. user time trade-off, where segmentation accuracy is measured with the Jaccard Index (overlap score) $J = \frac{P \cap GT}{P \cup GT}$ between the Predicted (P) and Ground Truth (GT) masks. The graph in Figure 3 plots the average Jaccard index relative to the amount of time users spent creating their annotations. Our experiments indicate that *Click'n'Cut* with experts converges more rapidly than the two graph-based approaches, but also that the resulting accuracy flattens out at a lower rate than either BPT or GrabCut. Moreover, Figure 3 shows that a crowd of non-expert users performs poorly when using *Click'n'Cut*, because of the high number of errors they make. When we filter the lowest-performing (i.e., those with an error rate above 20%) workers out of the crowd, the resulting accuracy becomes significantly higher. We postulate that this behavior can be explained by the fact that the spatial resolution of our solution based on combining object candidates is not as high as the regions or pixels used in GrabCut and BPTs.

Note that if we take the input from the crowd users (filtered by their gold standard performance) altogether, we obtain an average Jaccard Index of 0.83.

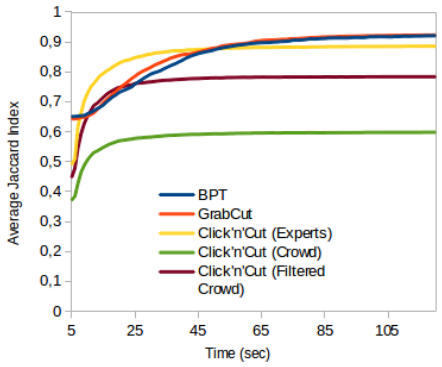

Figure 3: Average Jaccard index vs. user time.

4.2.3 Budget

The budget in our crowdsourced task can be approximated from two perspectives: *user time* and *money*.

Table 1 compares the necessary *user time* on *Click'n'Cut* with the data collected from other related publications. The first observation is the diversity of datasets used to solve interactive segmentation tasks prevents a direct comparison of the resulting values, with the exception of *Click'n'Cut* and [14], as detailed in Section 4.2.2. Our experiments also show that, given the same interface, experts tend to spend almost 50% more time than the crowd in generating the annotation, but that this higher dedication only produces a small increase in the quality of the segmentation. According to these data, the faster responsiveness of our system (already pointed in Section 4.2.2) seems to be confirmed with comparison with other solutions. This may be explained because the rest of crowdsourced systems are not exploiting any image processing algorithm to assist users in their task, which forces the user to manually draw the whole contour around the objects.

The comparison of cost in terms of *money* is more challenging, because it requires estimating the cost of the experts (a group that may contain undergraduate students, graduate students, research assistants and professors). The standard salary for a PhD grant in Ireland has been adopted as a reference, for a fairer comparison with [14], whose experts

	Dataset	User / image	Users type	Input	Avg. Time (sec.)	Average Jaccard
Chen [7]	KITTI	9	Experts	Tight Polygon	60	(Used as GT)
		?	Crowd	Tight Polygon	?	0.85-0.87
Lin [11]	Microsoft COCO	1	Crowd	Tight Polygon	79	(Used as GT)
Jain [10]	IIS+MSRC+CoSeg	5	Crowd	Box	7	-
		5	Crowd	Sloppy contour	20	-
		5	Crowd	Polygon	54	0.51-0.76
McGuinness [14]	BSDS (DCU subset)	20	Experts	Scribble	60-85	0.93
Click'n'Cut	BSDS (DCU subset)	**15**	**Experts**	**Click**	**32**	**0.89**
		20	**Crowd**	**Click**	**23**	**0.78 → 0.83**

Table 1: Comparison of *Click'n'Cut* with similar approaches, including average user time and best Jaccard.

where recruited in a research lab at Dublin City University. Using a 1,808 USD monthly wage as a reference, the annotation experiment involving 100 images and 20 experts would require a budget estimated in 377 USD. By contrast, our crowdsourcing campaign had a total cost of 130 USD, nearly three times cheaper than the experiment with experts.

5. CONCLUSION

This paper introduced *Click'n'Cut*, a web tool for interactive object segmentation designed for crowdsourcing tasks. The tool was tested with a crowdsourcing campaign to annotate images from publicly available datasets.

Experimental results are competitive with state-of-the-art interactive segmentation approaches in the literature. Despite some noisy traces, due to low-quality input from a few workers, the presented online interface has proven to be effective in collecting object segmentation results for crowdsourcing tasks, especially in terms of fast convergence to high quality results.

Moreover, when comparing the performance of experts and crowd workers on the same task using the same images, we have seen that using expert volunteers – at a cost of three times as much as crowdsourced workers – increased the average duration of the task by about 50%, while providing only a modest (less than 10%) increase in accuracy.

6. ACKNOWLEDGMENTS

The authors would like to thank Jordi Pont-Tuset for his valuable contributions, as well as all the experts who volunteered to run our experiments. This work has been partially supported by the Spanish Ministerio de Economía y Competitividad, under project TEC2013-43935-R.

7. REFERENCES

[1] T. Adamek. *Using contour information and segmentation for object registration, modeling and retrieval*. PhD thesis, Dublin City University, 2006.

[2] P. Arbelaez and L. Cohen. Constrained image segmentation from hierarchical boundaries. In *CVPR*, 2008.

[3] P. Arbeláez, J. Pont-Tuset, J. T. Barron, F. Marques, and J. Malik. Multiscale combinatorial grouping. In *CVPR*, 2014.

[4] S. Bell, P. Upchurch, N. Snavely, and K. Bala. Opensurfaces: A richly annotated catalog of surface appearance. *ACM TOG*, 32(4), 2013.

[5] Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary map; region segmentation of objects in n-d images. In *ICCV*, 2001.

[6] J. Carreira and C. Sminchisescu. Constrained parametric min-cuts for automatic object segmentation. In *CVPR*, 2010.

[7] L.-C. Chen, S. Fidler, A. L. Yuille, and R. Urtasun. Beat the mturkers: Automatic image labeling from weak 3d supervision. In *CVPR*, 2014.

[8] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2), 2010.

[9] X. Giró-i Nieto, M. Martos, E. Mohedano, and J. Pont-Tuset. From global image annotation to interactive object segmentation. *MTAP*, 70(1), 2014.

[10] S. D. Jain and K. Grauman. Predicting sufficient annotation strength for interactive foreground segmentation. In *ICCV*, 2013.

[11] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. *CoRR*, 2014.

[12] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, 2001.

[13] K. McGuinness and N. O'Connor. Improved graph cut segmentation by learning a contrast model on the fly. In *ICIP*, 2013.

[14] K. McGuinness and N. E. O'Connor. A comparative evaluation of interactive segmentation algorithms. *Pattern Recognition*, 43(2), 2010.

[15] A. Noma, A. B. Graciano, R. M. Cesar Jr, L. A. Consularo, and I. Bloch. Interactive image segmentation by matching attributed relational graphs. *Pattern Recognition*, 45(3):1159–1179, 2012.

[16] C. Rother, V. Kolmogorov, and A. Blake. "grabcut": interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.*, 23(3), Aug. 2004.

[17] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: A database and web-based tool for image annotation. *IJCV*, 77(1-3), 2008.

[18] P. Salembier and L. Garrido. Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval. *IEEE T. on Image Processing*, 9(4), 2000.

[19] A. Salvador, A. Carlier, X. Giro-i Nieto, O. Marques, and V. Charvillat. Crowdsourced object segmentation with a game. In *ACM CrowdMM*, 2013.