In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, June 2016.

Pull the Plug? Predicting If Computers or Humans Should Segment Images Supplementary Material

Danna Gurari Suyog Dutt Jain Margrit Betke Kristen Grauman

This document supplements Section 3 of the main paper. In particular, it includes the following:

1. Our segmentation initialization system illustrated by example (supplements **Section 3.2**).
2. Results illustrating the versatility of our segmentation initialization system for three segmentation tools (supplements **Section 3.2**).
3. Results illustrating the performance of our predicted initializations against two initialization baselines (supplements **Section 3.2**).
4. A variant of our initialization system for interactive segmentation tools which embed a shrinking bias (supplements **Section 3.2**).
5. Parallel results to Figure 3 in the main paper showing the benefit of our segmentation initialization system when using simulated human input (supplements **Section 3.2**).
6. Our fine-grained segmentation system illustrated by example (supplements **Section 3.3**).
7. Parallel results to Figure 5 in the main paper showing the performance of our system when evaluating human effort with respect to number of user clicks (supplements **Section 3.3**).
8. Methods to collect crowdsourced segmentations (supplements **Sections 3.2, 3.3**).

1. Our Segmentation Initialization System: Methods Illustration

Qualitative results exemplifying the steps of our segmentation initialization system (**Figure 1**). Each column is exemplifying how the system deploys the best option (segmentation with red boundary) per image from eight algorithm-generated options (one algorithm per row). As observed, different algorithms work well for different images. In addition, the six image examples together demonstrate how the predicted quality scores for all selected segmentations (i.e., row 1, Top Predicted Jaccard Score) are critical for subsequently ranking the 522 images in our study (Section 3.2 of main paper). The predicted scores automatically reveal which images to delegate to computers to segment. Specifically, given human annotation effort for K% of images for the 522 images, the system automatically distributes human effort to the K% of images where algorithms perform the worst (i.e, lowest Jaccard scores) and uses computer effort for the rest of images where the computer is predicted to have the best chance for success to create accurate *coarse segmentations*.

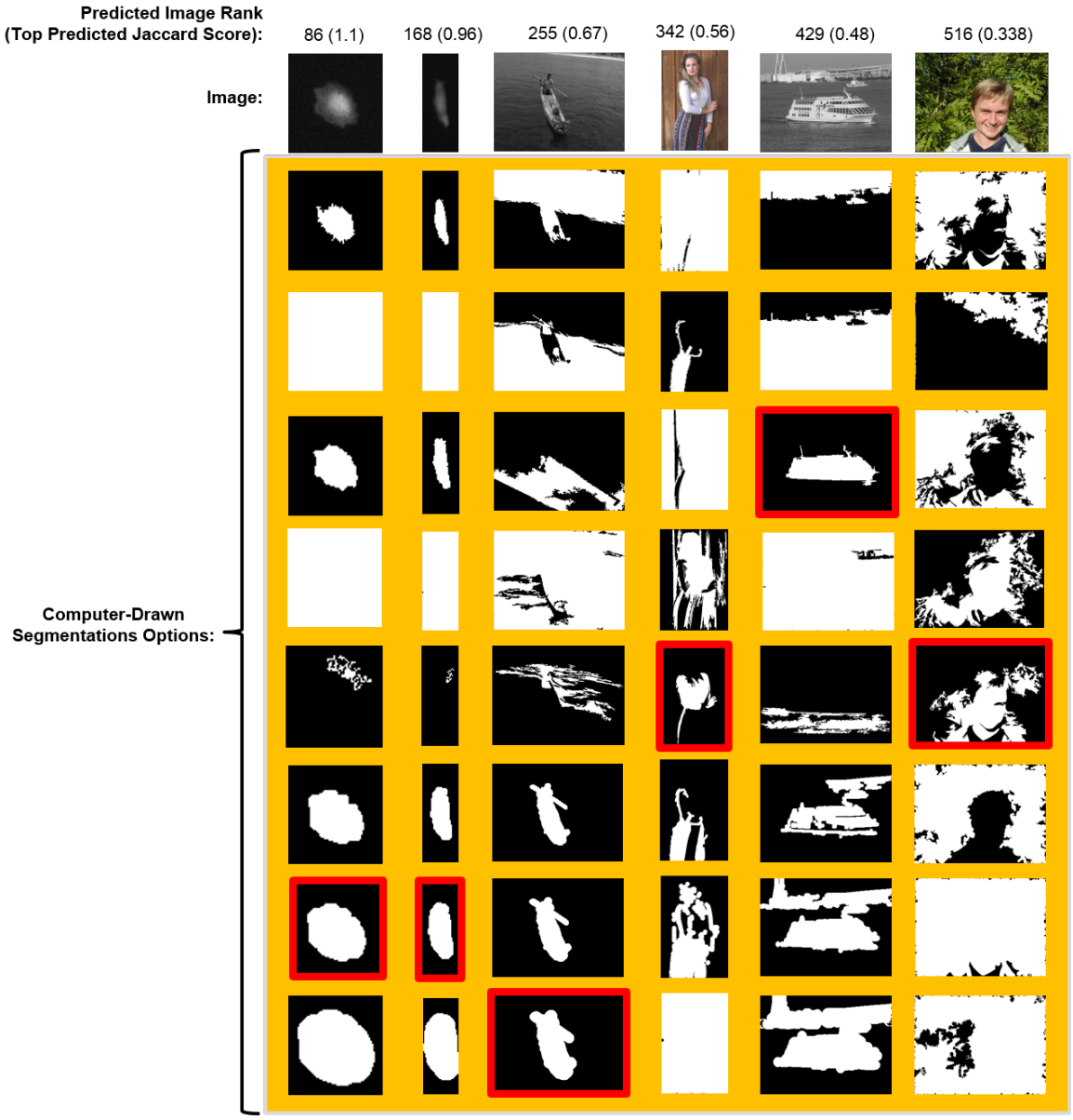

Figure 1. Our system intelligently pairs each image with the best option (segmentations with red boundaries) from eight options computed automatically by eight algorithms. Then, the system produces a relative ordering of images based on the predicted quality of all selected best computer-generated results for a given batch of images. Finally, the system uses this ordering to automatically decide which images should receive the allocated human budget.

2. Our Segmentation Initialization System: Results with Three Segmentation Tools

Qualitative results demonstrating the versatility of our system to initialize three different segmentation tools. Results are shown for biomedical (**Figure 2**) and everyday (**Figure 3**) images. Both figures show raw images (column 1) and the predicted input option from eight automatically generated options (column 2), followed by the resulting segmentation from the Grab Cut algorithm (column 3), Chan Vese level set algorithm (column 4), and Lankton level set algorithm (column 5). The ground truth segmentation is shown in column 6. In order to produce segmentations that resemble the ground truth, segmentation tools require sufficiently accurate initializations, which our system can produce automatically.

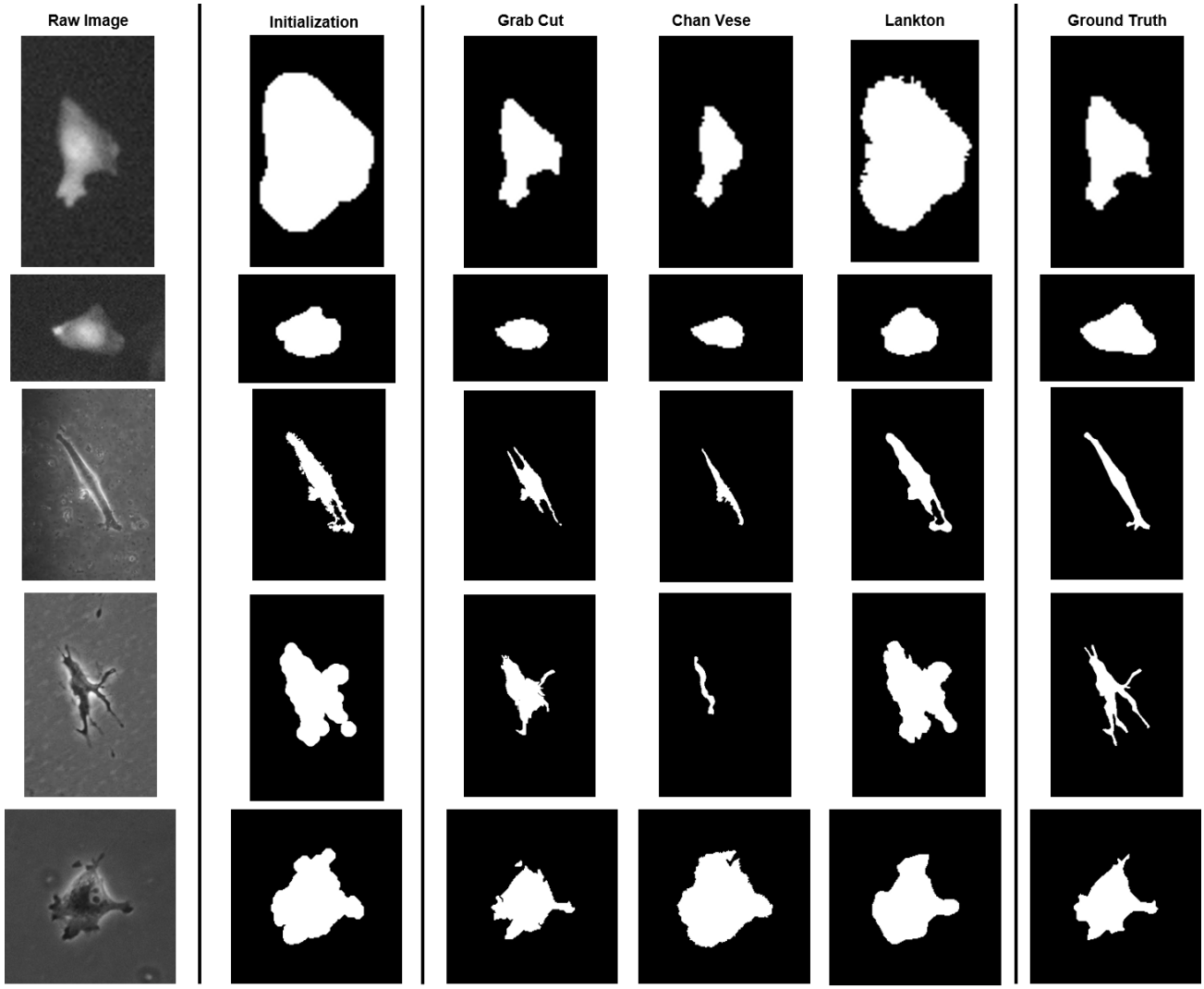

Figure 2. Sample results for the biomedical images (i.e., BU-BIL). Given the same initialization, the three segmentation tools can produce very similar segmentations in some cases (e.g., round cell shown in row 2) and dramatically different segmentations in other cases (e.g., spiculated cell shown in row 4).

Figure 3. Sample results for the everyday images (i.e., Weizmann and IIS). In some cases, a segmentation tool can perform well when using a low quality initialization, as observed for the image of the sheep (row 6, Grab Cut algorithm). In other cases, none of the three segmentation tools perform well when initialized poorly, as observed for the image of the person (row 4).

3. Our Segmentation Initialization System: Comparison to Baselines

Sample results for the Grab Cut algorithm when initialized with our fully-automated segmentation initialization system as well as two baselines (**Figure 4**). As observed in the "Successes" portion of **Figure 4**, the quality of segmentation results is higher with our intelligent selection approach than arbitrarily chosen initial segmentation estimates (Rectangle, Chance). As observed in the "Failures" portion of **Figure 4**, an initial segmentation estimate that does not fully contain the object of interest can lead to poor segmentation results. See the next Section for a variant of our approach that addresses this problem.

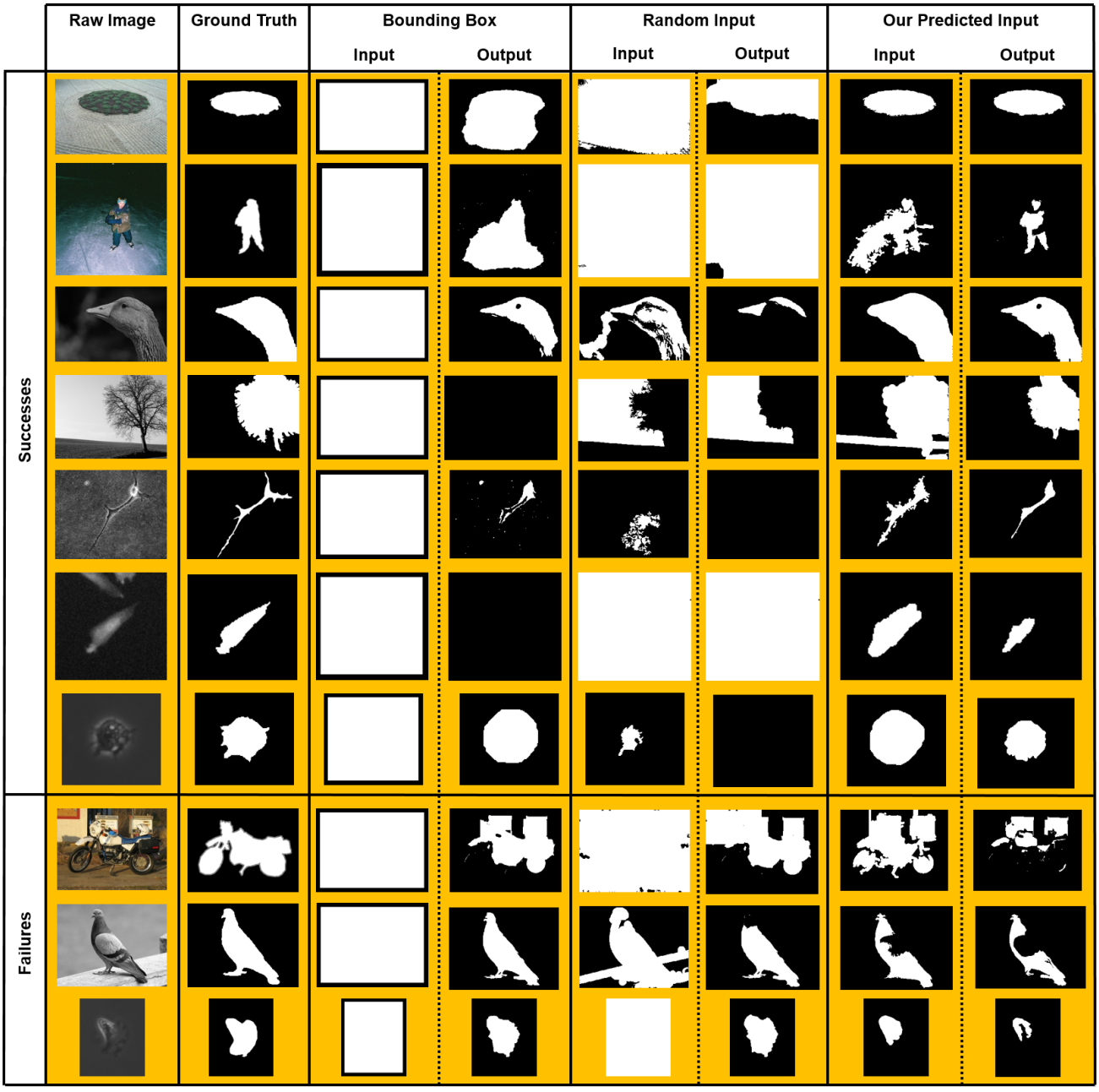

Figure 4. Performance of Grab Cut algorithm when it is paired with different initialization methods.

4. Our Segmentation Initialization System: Variant for Tools with a Shrinking Bias

Due to space constraints in the main paper, we discuss here the variant of our initialization system for use with segmentation tools that embed a shrinking bias. Specifically, while ideally the segmentation tools that refine *coarse segmentations* would support both shrinking and growing initial segmentations as needed (e.g., Chan Vese level set algorithm [2], Lankton level set algorithm [6]), a collection of segmentation tools exclusively shrink (or grow) initial segmentations [1, 7]. As discussed about the Grab Cut algorithm in the main paper (**Figure 3a**) and the previous Section of this document, the shrinking bias can lead to failures. To address the shrinking bias of Grab Cut, we leverage the bounding box of the automatically produced input from system instead. The results are shown in **Figure 5**, based on the simulated human input. The aim is to regularly enforce that the object is always fully contained in the predicted initialization. While there are clear improvements in the first 40% of human involvement from the bounding box of the predicted input, a valuable area for future work is to explore the impact of additional variants for "growing" the predicted input (e.g., dilation, convex hull, dilated convex hull).

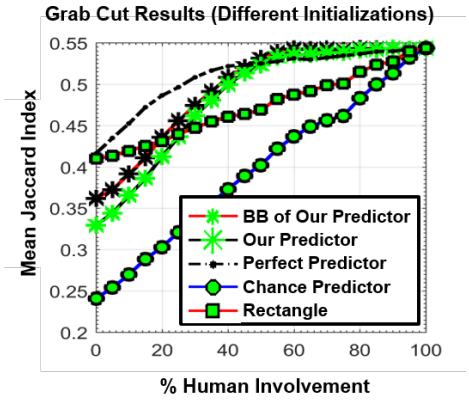

Figure 5. This plot is an augmented version of **Figure 3a** from the main paper. We include in this plot one additional curve to show the results from the bounding box of the automatically produced input from our system (BB of Our Predictor). This reveals a variant of our approach that may be better-suited for segmentation tools that embed shrinking biases.

5. Our Segmentation Initialization System: Analysis Using Simulated Input

While we show the results for budgeted human allocation based on real human input in **Figure 3** of the main paper, we show here the outcomes when using simulated human input (**Figure 6**). Specifically, following prior work [4], we simulate coarse human input by dilating the ground truth segmentations by 20 pixels. Although dilation may not perfectly capture how humans produce coarse segmentations in practice, it does offer insight into what one may expect when concerns about finding trustworthy humans are eliminated.

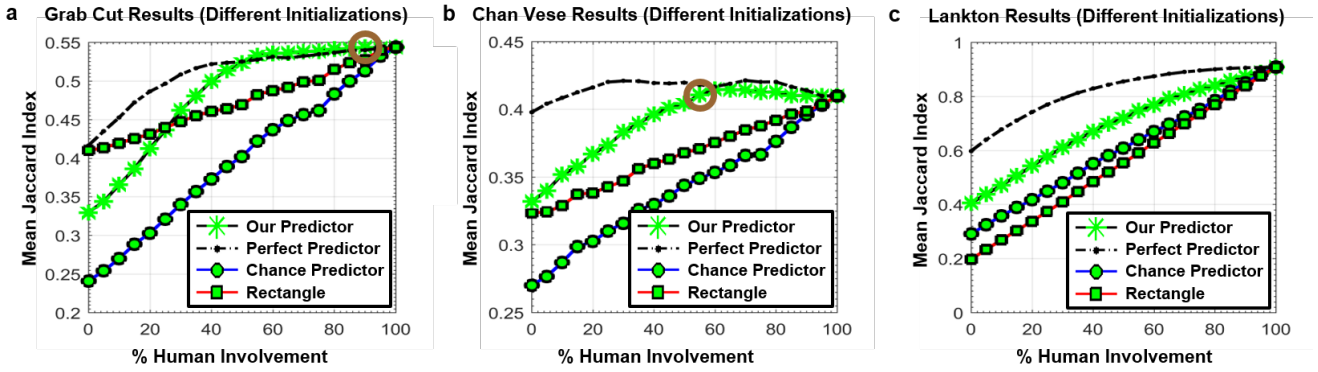

Figure 6. This result is parallel to Figure 3 of the main paper. The only difference is we use simulated human input here and crowdsourced coarse human input in the main paper. The outcomes relative to the baselines are the same in either case.

6. Our Fine-Grained Segmentation System: Methods Illustration

Qualitative results exemplifying the steps of our system to collect fine-grained segmentations (**Figure 7**). The six image examples together demonstrate how the predicted quality scores for all computer-generated segmentations are critical for ranking the 522 images in our study. As observed, the prediction system typically preserves the quality ordering between images in a batch. In addition, the predicted scores typically are close to the actual quality scores indicating how similar the computer-generated segmentation is to the ground truth segmentation with respect to the Jaccard index. Given human annotation effort for K% of images for the 522 images in our study (**Section 3.3** of main paper), the system automatically delegates that human effort to the K% of images where algorithms perform the worst (i.e, lowest predicted Jaccard scores) and uses computer effort for the rest of images where the computer is predicted to have the best chance for success to create accurate *fine-grained segmentations*.

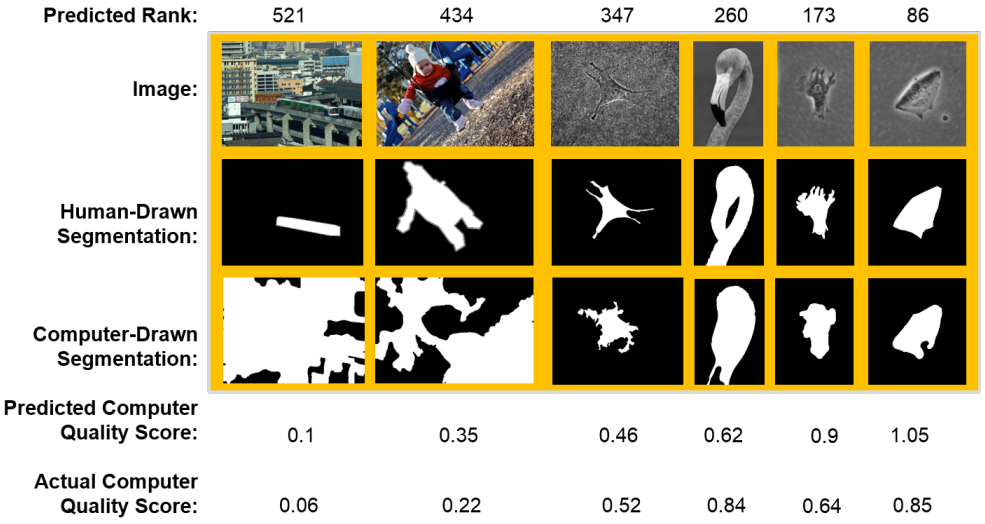

Figure 7. Our system produces a relative ordering of images based on the predicted quality of all computer-generated results for a given batch of images. This ordering is then used to intelligently decide when to have a human (i.e., ground truth segmentation) versus computer (i.e., Lankton level set algorithm initialized automatically) segment images. Images with worst rankings (lower predicted Jaccard scores for computer-generated segmentation) are selected first to have humans replace computers to create final, fine-grained segmentations.

7. Our Fine-Grained Segmentation System: Analysis Using Number of User Clicks

Our results here complement our analyses for budgeted human allocation in **Figure 5** of the main paper. Specifically, here we quantify human effort with respect to the number of user clicks needed to create the segmentation and there with respect to the time users took to complete each segmentation (**Figure 8**). For each image in BU-BIL and Weizmann, we use the average number of crowd worker clicks across five crowdsourced segmentations. For all images in IIS, we assign the average number of user clicks across all images in BU-BIL and Weizmann.

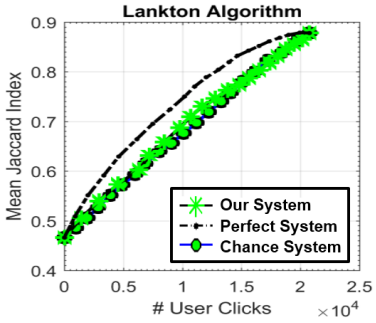

Figure 8. This result is parallel to Figure 5 of the main paper. The main difference is we quantify human effort with respect to number of user clicks to create the segmentation here and time to create the segmentation in the main paper. The outcomes relative to the baselines are the same in either case.

8. Prediction System Analysis

Due to space constraints, we excluded analyses of features for our prediction system from the main paper. We describe below which of the currently-used features (all based on binary masks) are most predictive. We report feature weights for all prediction models learned for "Ours - Single Set" experiments (**Table 1**). Shape-based features (S1-S3) most consistently offered the greatest predictive power amongst the additional boundary-based (B1-B2), coverage-based (C1-C2), and location-based (L1-L2) features. This suggests that prediction success is not due to overfitting to the size or location of objects. Features are described in **Section 2** of the main paper.

Table 1. Learned weights of predictive features.

Dataset:	BU-BIL	Weizmann	IIS	All
B1 - Std Dev of Boundary Distance to Centroid	0.0036	0.0016	0.0014	0.0035
B2 - Mean Boundary Distance to Centroid	-0.0006	-0.0007	-0.0008	-0.0005
C1 - Fraction of Bounding Box in Image	0.3742	0.3263	-0.1071	0.3561
C2 - Fraction of Object Pixels in Image	-1.0083	-0.5732	-0.2594	-0.9279
S1 - Shape Factor	0.4685	0.5877	0.6805	0.3614
S2 - Extent	-0.2196	-1.1679	-1.2402	-0.4161
S3 - Solidity	0.5517	1.4974	1.4257	0.9567
L1 - Normalized X-Coord	-0.1527	0	0.2309	-0.0508
L2 - Normalized Y-Coord	0.1044	0.1788	0.3342	0.3074

9. Crowdsourcing Segmentation Systems

All crowdsourced segmentations used for experiments in Sections 3.2 and 3.3 of the main paper are created by crowd workers from Amazon Mechanical Turk. For Section 3.2, we use the same crowdsourcing system employed in prior work [4] to collect coarse segmentations (sloppy contours). For Section 3.3, we leverage a collection of crowdsourcing methods. For Weizmann, we collect five segmentations per image using the image annotation tool LabelMe [8] and then use the pixel majority vote to create the final segmentation. We pay crowd workers $0.02 per segmentation. For BU-BIL, we use publicly-shared crowdsourced segmentations from Gurari et al. [3]. For IIS, we use publicly-shared crowdsourced segmentations from Jain et al. [4] and assign the average quality score from those segmentations to all images missing an annotation.

References

[1] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *IEEE Transactions on Image Processing*, 22(1):61–79, 1997. 6

[2] T. Chan and L. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–277, 2001. 6

[3] D. Gurari, D. Theriault, M. Sameki, B. Isenberg, T. A. Pham, A. Purwada, P. Solski, M. Walker, C. Zhang, J. Y. Wong, and M. Betke. How to collect segmentations for biomedical images? A benchmark evaluating the performance of experts, crowdsourced non-experts, and algorithms. *IEEE Winter conference on Applications in Computer Vision (WACV)*, page 8 pp., 2015. 8

[4] S. D. Jain and K. Grauman. Predicting sufficient annotation strength for interactive foreground segmentation. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1313–1320. IEEE, 2013. 6, 8

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1097–1105, 2012.

[6] S. Lankton and A. Tannenbaum. Localizing region-based active contours. *IEEE Transactions on Image Processing*, 17(11):2029–2039, 2008. 6

[7] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 3(309–314), 2004. 6

[8] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. LabelMe: A database and web-based tool for image annotation. *International Journal of Computer Vision*, 77(1–3):157–173, 2008. 8