International Journal of Computer Vision
https://doi.org/10.1007/s11263-019-01172-6

Predicting How to Distribute Work Between Algorithms and Humans to Segment an Image Batch

Danna Gurari[1] · Yinan Zhao[1] · Suyog Dutt Jain[1,3] · Margrit Betke[4] · Kristen Grauman[1,2]

Received: 31 May 2018 / Accepted: 25 February 2019

Abstract
Foreground object segmentation is a critical step for many image analysis tasks. While automated methods can produce high-quality results, their failures disappoint users in need of practical solutions. We propose a resource allocation framework for predicting how best to allocate a fixed budget of human annotation effort in order to collect higher quality segmentations for a given batch of images and automated methods. The framework is based on a prediction module that estimates the quality of given algorithm-drawn segmentations. We demonstrate the value of the framework for two novel tasks related to predicting how to distribute annotation efforts between algorithms and humans. Specifically, we develop two systems that automatically decide, for a batch of images, when to recruit humans versus computers to create (1) coarse segmentations required to initialize segmentation tools and (2) final, fine-grained segmentations. Experiments demonstrate the advantage of relying on a mix of human and computer efforts over relying on either resource alone for segmenting objects in images coming from three diverse modalities (visible, phase contrast microscopy, and fluorescence microscopy).

Keywords Foreground object segmentation · Interactive segmentation · Hybrid human–computer system · Crowdsourcing

1 Introduction

A common question people ask when needing to annotate their images is whether automated options are sufficient or they should instead bring humans in the loop to create accurate annotations. We explore this question for the task of demarcating object regions, i.e., creating *foreground object segmentations*. Foreground object segmentation is important for many downstream tasks including collecting measurements (features), differentiating between types of objects (classification), and finding similar images in a database (image retrieval). Our goal is to intelligently distribute segmentation work between humans and computers when human effort is limited.

Our work is partially inspired by the observation that fully-automated algorithms can produce high-quality foreground object segmentations when they are successful, yet their performance often is inconsistent on diverse datasets. This is because algorithms embed assumptions about how to separate an object from the background that are relevant for particular types of images, yet restrict their widespread applicability (Ballard 1981; Chan and Vese 2001; Lankton and Tannenbaum 2008; Otsu 1979; Rother et al. 2004). Consequently, the knowledge of when segmentation algorithms will succeed is currently a highly-specialized skill often resigned to computer vision experts or applications specialists who spent years studying the algorithms. Moreover, many researchers agree that there is not a one-size-fits-all segmentation solution. Thus, lay persons needing *consis-*

Communicated by Gang Hua.

✉ Danna Gurari
danna.gurari@ischool.utexas.edu

Yinan Zhao
yinanzhao@utexas.edu

Suyog Dutt Jain
suyog@utexas.edu

Margrit Betke
betke@bu.edu

Kristen Grauman
grauman@cs.utexas.edu

1 The University of Texas at Austin, 2317 Speedway, Stop, D9500, Austin, TX 78712, USA

2 Facebook AI Research, Menlo Park, USA

3 CognitiveScale, Austin, USA

4 Boston University, 111 Cummington Mall, Boston, MA 02215, USA

Published online: 16 March 2019

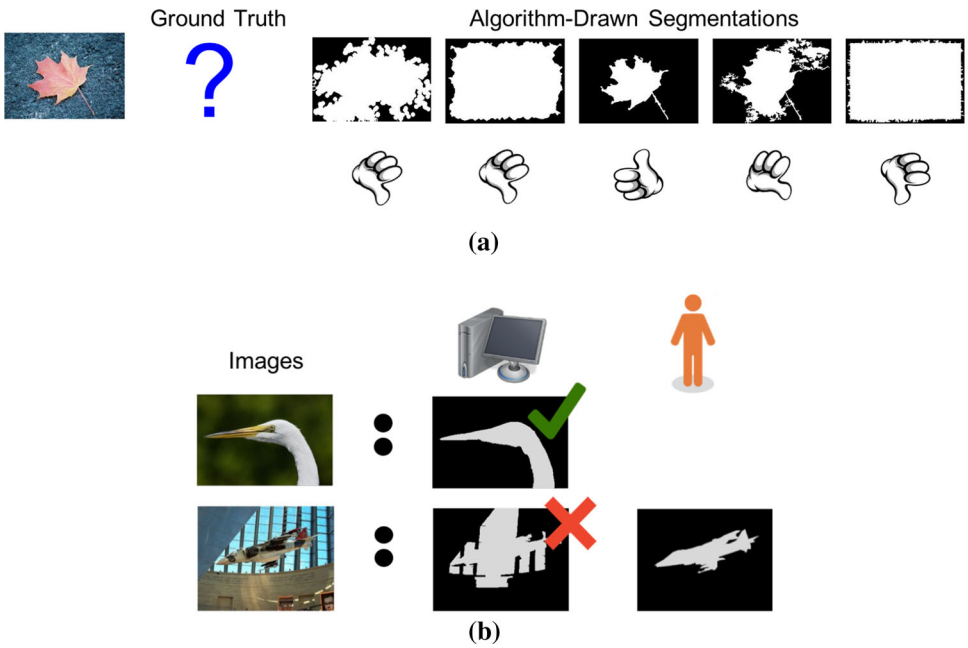


Fig. 1 We propose a task of predicting the quality of an image segmentation compared to the unseen ground truth in order to automatically (**a**) predict which among multiple algorithms will yield the highest quality segmentation and then (**b**) decide when to "pull the plug" on computers and use humans instead to create high quality segmentations

tently high quality segmentations currently face a brute force approach of reviewing all images with available algorithm-drawn segmentations to decide which algorithm is best-suited per image (Fig. 1a) and when to enlist human effort to re-annotate images because the best-suited algorithm produces a poor quality result (Fig. 1b).

Our work is also inspired by the observation that widely-used segmentation tools that rely on *initialization* are often inefficient because of their exclusive reliance on human input (Carlier et al. 2014; Grady et al. 2011; Gurari et al. 2014; Jain and Grauman 2013; Lempitsky et al. 2009; Rother et al. 2004; Wu et al. 2014). Specifically, humans create initial bounding boxes or coarse segmentations to localize the object of interest in every image. A motivation for leveraging human guidance per image is that a segmentation tool can only succeed when initializations are sufficiently close to the true object boundary (Jain and Grauman 2013). A weakness of relying on humans is that for numerous methods, including level set based methods (Bernard et al. 2009; Chan and Vese 2001; Lankton and Tannenbaum 2008; Li et al. 2008), users typically have to wait for minutes or more per image to validate whether the tool successfully converts their coarse input into high quality segmentations. Intuitively, one may expect that computers at times can create good enough segmentations to replace human initialization effort and so minimize human effort both for initialization and validation of the results. Still, lay persons typically lack the expertise to decide which images to distribute to computers.

We propose techniques to predict how to distribute annotation efforts between algorithms and humans for segmenting images. We address two novel tasks. First, we propose a system that intelligently allocates computer effort to replace human effort in order to create initial coarse object segmentations for refinement by segmentation tools. Second, we propose a system that automatically identifies images to have humans re-annotate from scratch by predicting which images the refinement methods segmented poorly. With both systems, a user provides a batch of images and indicates his/her available time for image annotation. In return, the system automatically decides for each image which algorithm will yield the best results and guides the user to only annotate images deemed to be most difficult for the available algorithms. More broadly, our systems could be exploited to efficiently create segmentations as input for downstream tasks, such as object recognition and tracking. We publicly share our code to support reproducing this work and future extensions (http://vision.cs.utexas.edu/HybridAlgorithmCrowdSystems/PullThePlug).

2 Related Work

Interactive *co-segmentation* methods address the issue of relying on human input to initialize segmentation tools for every image in a batch (Batra et al. 2010; Cui et al. 2008; Li et al. 2014). However, unlike our approach, these methods require that all images in the batch show related content (e.g., dogs). Moreover, interactive co-segmentation involves continual back-and-forth with an annotator to incrementally refine the segmentation. Avoiding a continual back-and-forth is particularly important for segmentation tools such as level set methods (Chan and Vese 2001; Lankton and Tannenbaum 2008) that take on the order of minutes or more per image to compute a segmentation from the initialization. We instead

recruit human input at most once per image and consider the more general problem of annotating unrelated, unknown objects in a batch.

Our aim to minimize human involvement while collecting accurate image annotations is shared by active learning (Settles 2010). Specifically, active learners try to identify the most impactful, yet least expensive information necessary to train accurate prediction models (Biswas and Parikh 2013; Settles 2010; Vijayanarasimhan and Grauman 2011). For example, some methods iteratively supplement a training dataset with images predicted to require little human annotation time to label (Vijayanarasimhan and Grauman 2011). Other methods actively solicit human feedback to identify features with stronger predictive power than those currently available (Biswas and Parikh 2013). Unlike active learners, which leverage human input at *training-time* to improve the utility of a single algorithm, our method leverages human effort at *test-time* to recover from failures by different algorithms.

Our novel tasks rely on a module to estimate the quality of computer-generated segmentations. Related methods find top "object-like" region proposals for a given image (Arbeláez et al. 2014; Carreira and Sminchisescu 2010; Endres and Hoiem 2010; Jain et al. 2017; Kohlberger et al. 2012). However, most of these methods are inadequate for ranking "object-like" proposals across a batch of images because they only return relative rankings of proposals per image (Endres and Hoiem 2010). Another method proposes an absolute segmentation difficulty measure based on the image content alone (Liu et al. 2011). However, this method does not account for the different performances that are observed from different segmentation tools when applied to the same image.

Our prediction framework most closely aligns with methods that predict the error/quality of a given algorithm-drawn segmentation in absolute terms (Carreira and Sminchisescu 2010; Kohlberger et al. 2012). In particular, we also perform supervised learning to train a regression model. However, prior work trained prediction models using segmentations created by a single popular algorithm (coming from the medical (Kohlberger et al. 2012) and computer vision (Carreira and Sminchisescu 2010) communities respectively). In contrast, our model is trained using a diversity of popular algorithms from different communities applied to images coming from three imaging modalities (visible, phase contrast microscopy, fluorescence microscopy). Specifically, we populate our training data with 14 algorithm-generated segmentation algorithms per image as well as ground truth data to capture a rich diversity of the possible quality of segmentations. Our approach consistently predicts well, outperforming a widely-used method (Carreira and Sminchisescu 2010) on four diverse datasets. Our experiments demonstrate the value of our prediction model for intelligently deciding which among multiple segmentation algorithms is preferable for each image.

More broadly, our work is a contribution to the emerging research field at the intersection of human computation and computer vision to build hybrid systems that take advantage of the strengths of humans and computers together. For example, hybrid systems combine non-expert and algorithm strengths to perform the challenging fine-grained bird classification task typically performed by experts (Branson et al. 2014; Wah et al. 2015). Another system decides how much human effort to allocate per image in order to segment the diversity of plausible foreground objects in a batch of images (Gurari et al. 2018). While our hybrid system design demonstrates the advantages of combining human and computer efforts, our work differs by deciding how to distribute work between more costly crowd workers and less expensive algorithms for the image segmentation task.

We initially presented these ideas at the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016 (Gurari et al. 2016). This work offers considerable redesigns to all methods which in turn yields significant improvements in our experimental results. Specifically, we propose an improved approach for predicting the quality of an algorithm-drawn segmentation by employing a larger training dataset (created using a larger collection of candidate algorithms) with an expanded feature set and ensemble regression model. We also introduce a hierarchical, two-stage prediction system that predicts which is the best algorithm per image to produce the initialization fed to the refinement-algorithm and then predicts the quality of the output from the refinement-algorithm in order to decide whether to solicit human input. Experimental results reveal our redesigned methods yield significant improvements for predicting the segmentation quality and producing high quality segmentations. We also expanded our experiments to explore the performance of our models and systems when using different feature sets and testing with different datasets in order to learn when, how, and why they succeed versus fail.

3 Segmentations by Humans or Computers?

We first describe two prediction systems for creating different levels of segmentation detail (Sect. 3.1). Then, we describe the module used by both systems to predict the quality of algorithm-generated segmentations (Sect. 3.2).

3.1 Batch Allocation of Humans and Computers

Our resource allocation framework predicts for each image in a batch whether the annotation should come from a human or computer. We call this framework *PTP* to reflect that the system predicts whether to "Pull The Plug" on computers and

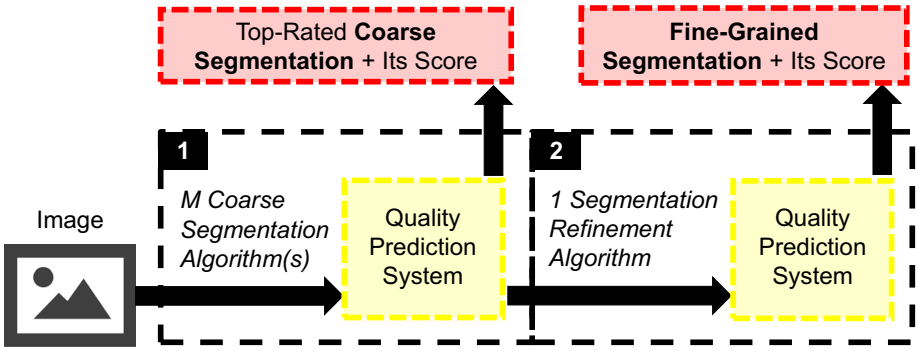


Fig. 2 Overview of the relationships between the coarse segmentation system, fine-grained segmentation system, and quality prediction module. Given an image, the coarse segmentation system applies multiple segmentation algorithms and outputs the top-predicted algorithm-generated result with its quality score. Given the coarse segmentation output by the coarse-segmentation system, the fine-grained segmentation system applies a refinement algorithm to it and outputs the resulting segmentation with its predicted quality score

solicit human effort for each image. We implement two *PTP* systems with the goals of creating coarse and fine-grained foreground object segmentations respectively. We examine the value of our systems with segmentation tools that require initialization. These tools are well-suited for studying both systems because they require coarse object segmentation input and aim to output high quality, fine-grained object segmentations. Fig. 2 summarizes the relationship between the coarse segmentation system, fine-grained segmentation system, and the segmentation quality prediction method.

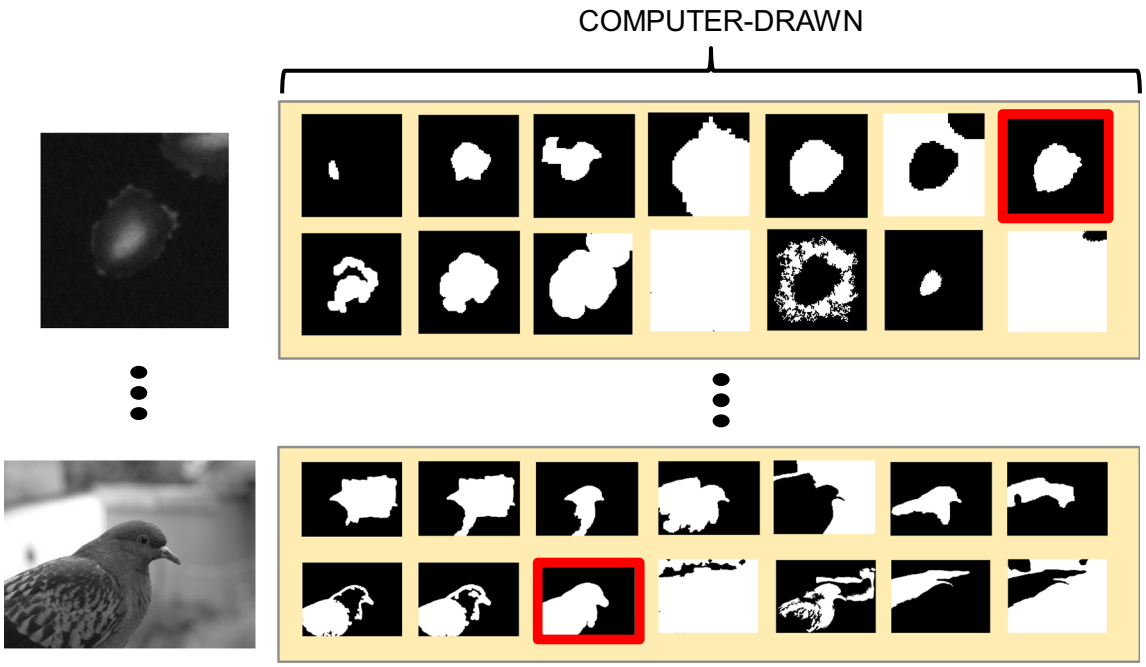
Like existing interactive segmentation methods, we assume the user is interested in a primary foreground object (Carlier et al. 2014; Grady et al. 2011; Lempitsky et al. 2009; Rother et al. 2004; Wu et al. 2014). That is, there is a primary object of interest that the user wishes to isolate from the background. Foreground object segmentation is therefore distinct from natural scene segmentation, where methods aim to segment all objects present in the image or delineate their boundaries or primary contours (Arbelaez et al. 2011; Everingham et al. 2010; Martin et al. 2001).

3.1.1 Coarse Segmentation: Computer or Human?

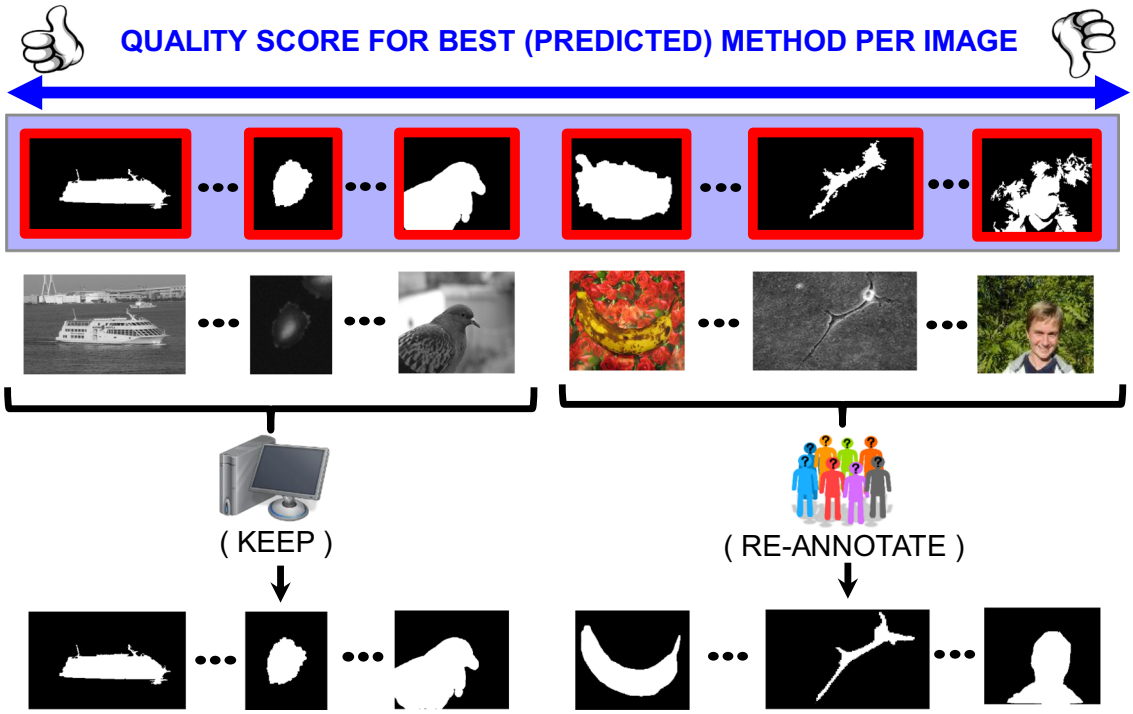
Our first system automatically decides when to delegate the task of creating *coarse segmentations* refined by segmentation tools to computers in an effort to improve upon today's status quo of relying exclusively on human input (Batra et al. 2010; Cui et al. 2008; Li et al. 2014). We intentionally designed the system to be agnostic to the particular refinement segmentation tool. We implemented the system to run a refinement segmentation tool exactly once per image with one input since some tools are time-consuming to run (Chan and Vese 2001; Lankton and Tannenbaum 2008), requiring minutes or more to refine a single initialization. In the interest of increasing the chance of computer success, our system predicts which from a larger list of 14 computer-generated results is best-suited to create the coarse segmentation input per image. Then, our system decides for each image whether to deploy the top-rated computer-generated coarse segmentation versus instead enlist a human to produce the initial coarse segmentation.

Figure 3 exemplifies the six steps of our initialization system. First, the system collects 14 algorithm-drawn foreground segmentations per image described in Sect. 3.2, then predicts the quality of each candidate segmentation using our proposed prediction system discussed in Sect. 3.2, and then deploys the top-scoring option as the computer choice (Fig. 3a). Next, all images are sorted based on the selected computer choices, from highest to lowest predicted quality scores, and then the system allocates the available human budget to create coarse segmentations for the allotted number of images with the lowest predicted quality scores (Fig. 3b). In other words, the system relies on human effort only for the images where the computer is predicted to have the worst chance to create accurate coarse segmentations. Finally, the system feeds all coarse segmentations created by humans and computers to the segmentation tool of interest for refinement.

For the candidate algorithms chosen to produce computer-drawn coarse segmentations, we were motivated to employ fully-automated methods that consistently yield high-quality segmentations across the various image modalities investigated in this paper (visible, phase contrast microscopy, fluorescence microscopy). Towards this aim, we rely on 14 variants of six algorithms discussed in current literature for foreground object segmentation both for the mainstream computer vision community (Arbeláez et al. 2014; Carreira and Sminchisescu 2010; Liu et al. 2011) as well as the biomedical imaging community (Chittajallu et al. 2015; Glenn et al. 2015; Maitra et al. 2012). Specifically, included are the top-ranked segmentations output by two region proposal methods: multiscale combinatorial grouping (i.e., MCG) (Arbeláez et al. 2014), and constrained parametric min-cuts (i.e., CPMC) (Carreira and Sminchisescu 2010). Also included is a salient object segmentation method



(a)



(b)

Fig. 3 Overview of the segmentation initialization system. Given a batch of images, (**a**) the system automatically pairs each image with the resulting segmentation from multiple algorithms that is predicted to be the highest quality (highlighted in red). Shown are the 14 options per image in the following order: top-3 MCG proposals (Arbeláez et al. 2014), top-3 CPMC proposals (Carreira and Sminchisescu 2010), salient object segmentation (Liu et al. 2011), Hough Transform with circles from radii with 3, 5, and 10 (Ballard 1981), adaptive thresholding and its complement, and finally Otsu thresholding and its complement (Otsu 1979). (**b**) Then, the system produces a relative ordering of all images based on the predicted quality of all selected best computer-generated results. Finally, the system automatically allocates the available human annotation budget to images with the predicted lowest quality segmentations and keeps the automated results for the remainder of the images (Color figure online)

which establishes a segmentation using a combination of local, regional, and global statistics for an image (Liu et al. 2011). Finally, our system produces segmentations using three popular biomedical image algorithms (Chittajallu et al. 2015; Glenn et al. 2015; Maitra et al. 2012): Hough Transform with Circles (Ballard 1981), Otsu Thresholding (Otsu 1979), and adaptive thresholding. We increase the number of options by employing the following variants of the aforementioned methods; i.e., using the top three region proposals per method (Arbeláez et al. 2014; Liu et al. 2011), augmenting the image complement of the segmentation for both thresholding methods (Otsu 1979), and employing different radius values for the algorithm (Ballard 1981) (i.e., 3, 5, 10). Our system then post-processes each binary mask by filling all holes and keeping only the largest object.

3.1.2 Fine-Grained Segmentation: Computer or Human?

A related yet more challenging task is predicting whether a computer-generated segmentation captures the fine-grained details describing a true object region or whether humans should instead segment images from scratch. Whereas the first system above elicits coarse human input to initialize a segmentation tool, we next propose a second system that elicits fine-grained human input to replace segmentation tools when they segment images poorly. The motivation of the system design is to offer a better solution than today's status quo of humans reviewing all images with associated segmentations to spot algorithm failures.

This system consists of five key steps to segment a given batch of images. First, a coarse segmentation is automatically generated for every image using the aforementioned *Coarse Segmentation* system to choose the best computer-drawn segmentation per image from 14 options (see Fig. 2, part 1). Then, each coarse segmentation is refined by a segmentation tool to produce a final segmentation for each image. Next, the prediction system discussed in Sect. 3.2 is applied again to estimate the quality of each resulting segmentation (see Fig. 2, part 2). Then, the system sorts all images from highest to lowest predicted quality scores for the resulting segmentations. Finally, the system allocates the available human budget to create fine-grained segmentations for the allotted number of images with the lowest predicted quality scores.

3.2 Predicting Segmentation Quality

Embedded in both the *Coarse* and *Fine-Grained* segmentation systems above is a module which automatically predicts the similarity of a given segmentation to an unseen ground truth segmentation (Fig. 4a). We propose as our prediction framework a regression model in order to capture that algorithms can produce segmentations that range in quality from complete failures to nearly perfect.

3.2.1 Training Instances

We aim to populate our training data with segmentation masks that reflect a large, relatively balanced number of examples for each segmentation quality from the range of possible segmentation qualities. Towards this aim, we choose segmentation masks that capture the transition of segmentation quality from perfect (i.e., ground truth), to reasonable human mistakes (i.e., manipulated ground truth), to a variety of possible failure behaviors (i.e., various algorithms). Accordingly, for each image, our system produces multiple training examples derived from the human-drawn ground truth as well as 14 algorithm-drawn segmentations.

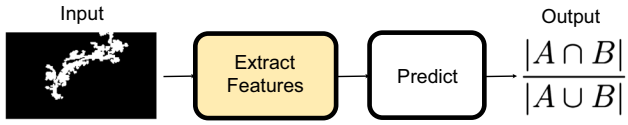
For algorithm-drawn segmentation masks, we employ the same 14 methods used in the *Coarse Segmentation* system described in the Sect. 3.1, which includes region proposals (Arbeláez et al. 2014; Carreira and Sminchisescu 2010), salient object segmentation (Liu et al. 2011), and popular biomedical image segmentation algorithms (Chittajallu et al. 2015; Glenn et al. 2015; Maitra et al. 2012). The variety of failure behaviors produced by the different algorithms are exemplified in Fig. 4b, columns 4–6.

Given that the training data may be insufficiently populated with higher-scoring segmentations (if all algorithms consistently fail), our system augments three binary masks based on the ground truth segmentations. The system uses the ground truth directly in order to capture during training the appearance of a perfect segmentation. Our system also dilates and erodes the ground truth binary mask by three pixels to simulate a slightly under-segmented and over-segmented segmentation respectively where fine details may get smoothed out or chopped off (e.g., Fig. 4b; columns 3–4).

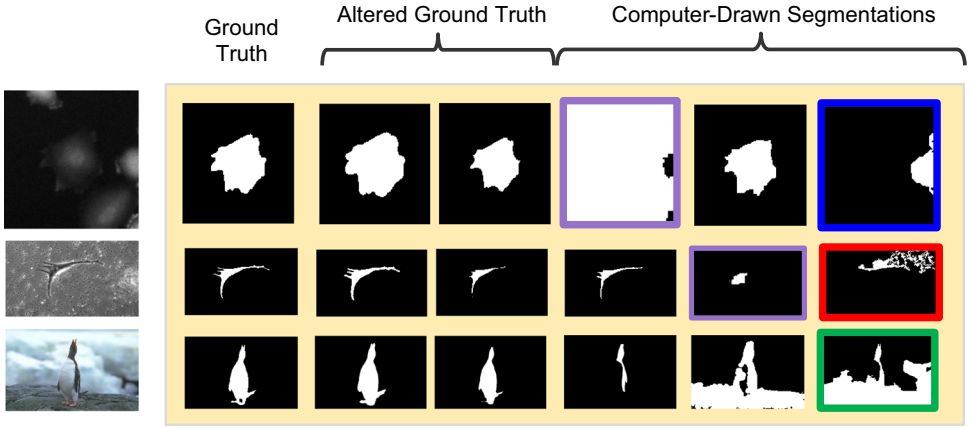
3.2.2 Training Data: Features

Next, our motivation is to use knowledge about algorithm behavior on everyday and biomedical images to choose predictive features. We take advantage of the observation that the chosen algorithms sometimes fail big when they fail, manifesting appearances unlike what one would expect from widely meaningful object shapes (Fig. 4b). We propose nine features that describe the binary segmentation mask to capture these failure behaviors. We also consider image descriptors based on convolutional neural networks (i.e., CNNs). We hypothesize that, in aggregation, these features may account for objects of different shapes, sizes, and appearances. We describe these features below.

Segmentation boundary When algorithms fail, resulting segmentations often have boundaries characterized by an



(a)



(b)

Fig. 4 (**a**) At test time, given a segmentation of an image, our system predicts a score indicating its similarity to the unobserved ground truth. (**b**) To train our prediction system, we employ images showing foreground objects from three diverse domains: fluorescence microscopy (row 1), phase contrast microscopy (row 2), and visible spectrum imaging (row 3). Examples of algorithm-generated results illustrate our training data captures a wide range of segmentation outcomes spanning from perfect (i.e., ground truth) to various failure behaviors (i.e., from different algorithms). As shown, cues of algorithm failures are observed in the segmentation's boundary (highlighted in red), compactness (highlighted in green), location (highlighted in blue) and its image coverage (highlighted in purple) (Color figure online)

abnormally large proportion of highly-jagged edges. We implement two boundary-based features to capture this observation. We compute the *mean* and *standard deviation of the Euclidean distance of every point on the segmentation boundary to the centroid*. The boundary is defined as all pixels on the exterior of the object in a binary mask using an 8-connected neighborhood. The centroid is defined as the center of mass of the segmentation in the binary mask.

Segmentation compactness When algorithms fail, segmentations often are not compact. We designate three features to capture this observation. Two measures compute the coverage of segmentation pixels within a bounding region. *Extent* is defined as the ratio of the number of pixels in the segmentation to the number of pixels in the area of the bounding box. *Solidity* is defined as the ratio of the number of pixels in the segmentation to the number of pixels in the area of the convex hull. We also compute the *shape factor* to capture the circularity of the segmentation since a pure circle is a good measure to indicate highly compact objects. It is defined as the ratio of region area A to a circle with the same perimeter P: $\frac{4\pi A}{P^2}$.

Location of segmentation in image When algorithms fail, resulting segmentation regions often lie closer to the edges of images. This observation stems in part from the center bias of many existing datasets. We compute the *normalized x* and *y centroid coordinates* of the segmentation centroid in the image to capture this observation. Specifically, we compute the x value of the center of mass divided by the image width and y value of the center of mass divided by the image height.

Coverage of segmentation in image When algorithms fail, resulting segmentations often cover abnormally large and small areas in the image. We implement two features to capture this observation. First, we compute the *fraction of pixels in the image that belong to the segmentation*. Second, we compute the *fraction of pixels in the image that belong to the bounding box of the segmentation*.

Image-based CNN features The above features capture elements likely to be informative for the task, based on domain knowledge of the binary segmentation mask. As a counterpoint, we also consider feature vectors extracted from three off-the-shelf CNN architectures in order to describe the intensity values of the segmentation mask. Specifically, we use CNN features coming from three classification systems which were pre-trained on ImageNet: AlexNet (Krizhevsky et al. 2012), VGG (Simonyan and Zisserman 2014), and ResNet (He et al. 2016). For AlexNet, we use the last fully connected layer to create a 4096-dimensional vector. For VGG, we use the fc7 layer to create a 4096-dimensional vec-

Table 1 Characterization of studied datasets to reveal the diversity of image content with respect to object area (# pixels), centroid location (X Loc, Y Loc), shape (Sect. 3.2; shape factor), and coverage in image ($\frac{\text{FG Area}}{\text{Image Area}}$) as well as image texture (FG Var, BG Var = variance of Laplacian values for object and background pixels respectively)

	BU-BIL		Weizmann		IIS		MSRA10K	
	μ	σ	μ	σ	μ	σ	μ	σ
Area	7927	13,109	24,315	16,815	40,119	41,387	26,235	12,489
X Loc	126	129	146	29	251	80	186	52
Y Loc	115	106	158	61	223	63	171	44
Shape	0.48	0.25	0.41	0.2	0.4	0.2	0.50	0.24
$\frac{\text{FG Area}}{\text{Image Area}}$	0.12	0.04	0.27	0.14	0.19	0.12	0.22	0.10
FG Var	54	51	1663	1271	2227	1909	1292	1244
BG Var	28	36	540	835	1568	1521	587	829

tor. For ResNet, we use the pool5 layer after global average pooling to obtain a 2048-dimensional vector. We compute these feature representations using the image patch created by using the bounding box of the segmentation.

See Sect. 4 for an analysis of the variability of several of the mask-based cues measured for objects observed within diverse datasets. This analysis highlights the variability and biases available in a range of unrelated datasets.

3.2.3 Training Data: Labels and Regression Model

To create each output label, the system computes a score indicating the quality of each training instance segmentation. We use the Jaccard index (i.e., intersection over union, IOU) which indicates the fraction of pixels that are in common to the training instance and ground truth segmentation (i.e., $\frac{|A \cap G|}{|A \cup G|}$).

For our model, we train a regression tree ensemble with the aforementioned training data to predict the quality of a given segmentation of an image. This model is trained to learn the unique weighted combinations of the features that each of a collection of regression trees applies to make a prediction. This offers a relatively fast, minimally data hungry approach that can be used with many hardware platforms, making it accessible to niche communities for easy use and re-training for their specific algorithms and datasets. We employ 25 trees and train by sampling one third of the predictive variables per decision split, sampling training examples with replacement, and requiring a minimum of five examples per tree leaf.

4 Experiments and Results

We conduct studies to analyze the reliability of our prediction framework and its value for deciding when to target computers or humans to segment images.

Datasets We evaluate our methods on four datasets that represent three imaging modalities: Boston University Biomedical Image Library (BU-BIL:1-5) (Gurari et al. 2015) includes 271 gray-scale images coming from three fluorescence microscopy image sets and two phase contrast microscopy image sets, Weizmann (Alpert et al. 2007) consists of 100 grayscale images showing a variety of everyday objects, Interactive Image Segmentation (Gulshan et al. 2010) (IIS) contains 151 RGB images showing a variety of everyday objects, and MSRA10K (Cheng et al. 2014) contains 10,000 RGB images showing a variety of everyday objects. Each dataset contains human-drawn segmentations that serve as pixel-accurate ground truth segmentations.

Together, the four datasets exhibit large variability with respect to object and image properties (Table 1). For example, the object size is over five times larger in IIS (i.e., 40,119 pixels) than in BU-BIL (i.e., 7927 pixels). The object consumes more than two times the area of the image in Weizmann (i.e., 0.27) than in BU-BIL (i.e., 0.12). Moreover, there is rich diversity of object appearances within each dataset, as revealed by large $\boldsymbol{\sigma}$ values. For example, there is a large Shape $\boldsymbol{\sigma}$ for all datasets. Additionally, the variability of object texture (i.e., FG Var $\boldsymbol{\sigma}$) is relatively large for all datasets. Furthermore, our analysis suggests that image backgrounds can be complicated and/or cluttered (i.e., large BG Var $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$). The observed diversity of dataset characteristics is important to ensure our method is challenged to learn generic cues predictive of segmentation failure.

4.1 Quality Prediction for Algorithm Set

We first analyze the predictive power of our proposed framework to automatically estimate the quality of foreground object segmentations (Sect. 3.2).

Baseline We compare our method to the *CPMC* (Carreira and Sminchisescu 2010) approach that also can predict the quality of any given object segmentation. Specifically, it predicts a Jaccard score per segmentation. This baseline stresses generality by learning statistics typical for real world objects. The method learns to predict Jaccard scores on everyday images using a combination of shape and intensity-based features. We use publicly-available code. We do not compare against methods that return a relative ranking of proposal regions per

Table 2 Comparison of CPMC (Carreira and Sminchisescu 2010) with our method for predicting the Jaccard score indicating the quality of a foreground segmentation. We report scores for our method learned with cross-set training ("C-Ours") and single-set training ("S-Ours") when using mask features alone ("-M"), intensity features alone ("-I"), as well as both mask and intensity features. For intensity features, we consider three CNN options: AlexNet (Krizhevsky et al. 2012), VGG (Simonyan and Zisserman 2014), and ResNet (He et al. 2016). We conduct experiments with four datasets. Higher correlation coefficient (CC) scores and lower mean absolute error (MAE) scores are better

	BU-BIL		Weizmann		IIS		MSRA10K	
	CC	MAE	CC	MAE	CC	MAE	CC	MAE
C-CPMC (Carreira and Sminchisescu 2010)	0.18	0.30	0.27	0.30	0.23	0.32	0.61	0.25
C-Ours	0.63	0.21	0.61	0.23	0.50	0.26	**0.62**	**0.23**
C-Ours-I	−0.10	0.33	0.40	0.27	0.41	0.29	0.49	0.26
C-Ours-M	0.66	0.19	**0.65**	**0.21**	**0.56**	**0.23**	0.59	0.22
S-Ours	0.87	0.10	**0.85**	**0.15**	0.83	0.16	**0.86**	**0.12**
S-Ours-I	0.84	0.13	0.81	0.16	**0.84**	**0.15**	0.80	0.15
S-Ours-M	**0.88**	**0.09**	0.79	0.16	0.73	0.19	0.78	0.15

Bold values are used to identify the model with the best performance

image (e.g., Endres and Hoiem 2010), because they are inadequate for ranking segmentations across a batch of images.

Evaluation metrics We evaluate each prediction model using Pearson's correlation coefficient (CC) and mean absolute error (MAE). CC indicates how strongly correlated predicted scores are to actual Jaccard scores for all foreground object segmentations evaluated. Values range between +1 and −1 inclusive, with values further from 0 indicating stronger predictive power. MAE is the average size of prediction errors, computed as the mean absolute difference between all predicted and actual Jaccard scores.

Ours: cross-dataset generalization To minimize concerns that prediction successes may be due to over-fitting to the statistics of a particular dataset, we first evaluate how well our prediction models trained on three of the datasets perform on the fourth dataset.[1] We enrich our analysis by also examining the predictive performance of our models when trained and tested exclusively with the mask-based and intensity-based features respectively. Table 2 shows our results when employing both mask-based and intensity-based features (row 2), intensity-based features alone which is the CNN features described in Sect. 3.2.2 (row 3), and mask-based features alone which is all features described in Sect. 3.2.2 except for the Intensity features (row 4). For clarity in presenting the results, we only show results for the overall top-performing CNN feature, based on testing both with mask-based features and alone, from the three evaluated options: AlexNet (Krizhevsky et al. 2012).

Overall, our approach performs well, as indicated by high CCs and low MAEs (Table 2, row 2). The significant improvement of our approach over CPMC on the biomedical images (e.g., row 2 versus 1 with CC of 0.63 versus 0.18) shows it is successful even when trained on completely disjoint datasets—what the system learned on everyday images (Weizmann, IIS, MSRA10K) can successfully be leveraged on biomedical images (BU-BIL). This is possibly because algorithms tend to create binary masks that have consistent properties at various levels of success and failure severity, regardless of the dataset. Our approach also yields improvements over CPMC on the everyday images (Weizmann, IIS, MSRA10K), highlighting a potential value of populating training data with images from different modalities (e.g., biomedical images) to promote learning generic algorithm behavior rather than over-fitting to properties of a particular dataset.

We observe that most of the predictive power of our model stems from mask-based features. Mask-based features (Table 2, row 4) perform better than variants of our model that employ intensity-based features (rows 2–3) for all but one dataset, when comparing CC and MAE scores. This reveals a plausible limitation that intensity features do not generalize as well for different objects observed in images captured with different image acquisition technologies (e.g., microscopes) and parameters (e.g., lighting). This hypothesis is supported by the observation that relying on the off-the-shelf CNN feature alone yields negligible predictive power for the biomedical images (Table 2, row 3). Moreover, we hypothesize the intensity-based features leads to high MAE values because of an accumulation of errors from using a high dimensional feature space. Our findings demonstrate the characteristics of segmentation errors are robustly and sufficiently learned from a small set of features describing the binary mask alone and remain relevant across domains.

Ours: single-dataset analysis Having demonstrated the advantage of our approach over an existing state-of-art base-

[1] To afford similar contributions of each dataset, we randomly sample 2000 segmentations for the MSRA10K dataset.

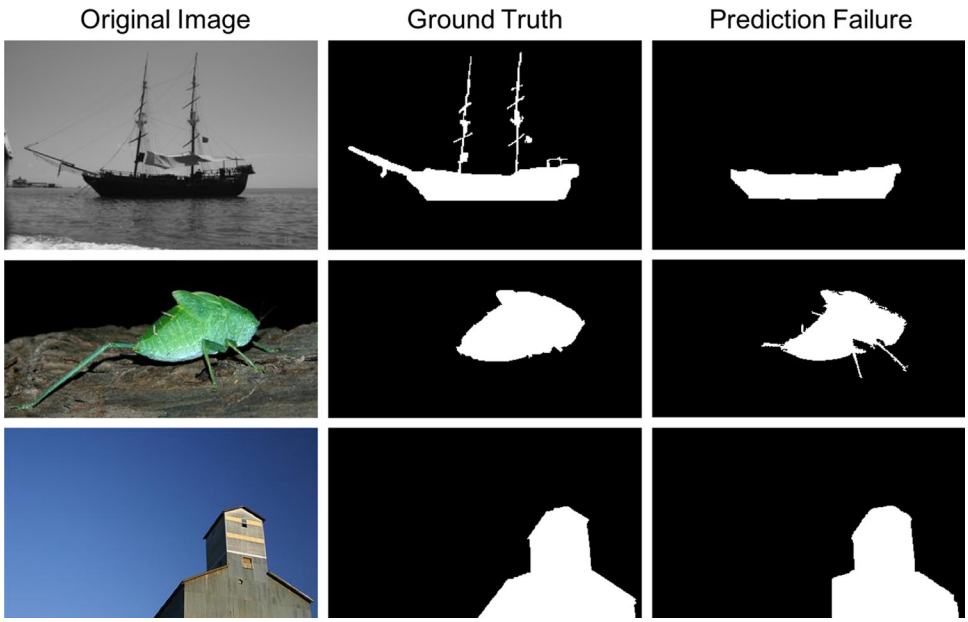


Fig. 5 Examples of images for which our prediction system makes inaccurate predictions. Shown is the original image in the left column, ground truth foreground segmentation in the middle column, and an algorithm-drawn segmentation where our prediction module fails in the right column

line (i.e., CPMC) for cross-dataset tests, we next examine the performance gain of using our model when evaluating our prediction framework for each dataset independently (i.e., Weizmann, IIS, BU-BIL, MSRA10K). Specifically, we train and test using ten-fold cross-validation on each dataset separately. We again enrich our analysis by examining the predictive performance of our models when also trained and tested exclusively with the mask-based and intensity-based features respectively. Table 2 shows results from employing all features (row 5), intensity-based features alone (row 6), and mask-based features alone (row 7).

As to be expected, we consistently observe further performance improvements when focusing on individual datasets (rows 5–7) rather than across datasets (rows 2–4); e.g., CC improves from $\sim$0.60 across each dataset to $\sim$0.85 (Table 2, row 2 versus 5) when using both mask-based and image-based features. Interestingly, different sets of features are more predictive for different datasets. For example, the most predictive features are mask-based for BU-BIL, intensity-based for IIS, and the combination of both for Weizmann and MSRA10K. We hypothesize the predictive features stem from the distinct biases of individual datasets. For example, we hypothesize mask-based features matter more for BU-BIL because the dataset has relatively stronger shape-based biases; e.g., as observed in Table 1, objects in BU-BIL exhibit relatively little variation in image coverage (i.e., $\boldsymbol{\sigma} = 0.04$).

Our findings also highlight the impact of using different amounts of data for training. Specifically, when comparing the performance of utilizing 10,000 images in MSRA10K versus $\sim$100–300 images in BU-BIL, Weizmann, and IIS for all feature combinations, we observe similar outcomes; i.e., CC scores range from 0.78 to 0.86 for MSRA10K which is slightly worse than the findings for BU-BIL and Weizmann and slightly better than the findings for IIS (Table 2, rows 5–7). This suggests that smaller datasets are sufficiently large for learning predictive cues that generalize.[2] We also show qualitative results that illustrate some failure cases of our top-performing single-dataset prediction module, which employs mask-based feature vectors extracted from the AlexNet architecture (Fig. 5). For all the examples, the predicted segmentation captures the main body of the object but receives low scores from the prediction module. While the top and bottom examples are missing some object parts, the middle example appears to be penalized for capturing the highly-jagged edges on the boundary (excluded from the ground truth) despite that it still successfully captures the main body of the object.

Overall, our findings show it is possible to predict the quality of an image segmentation in absolute terms for a diversity of data spanning everyday and biomedical images. As will be shown in the following sections, this capability offers exciting implications towards deciding which among multiple algorithms to choose to create the highest quality segmentations and deciding how to distribute effort between computers and humans to create high quality segmentations for a batch of images.

[2] For the one dataset large enough to train a deep model, MSRA10K, we find that fine-tuning off-the-shelf CNNs (namely, AlexNet, VGG, and ResNet) yields similar or worse performance than the other models tested in our experiments, including those using the frozen CNN features without fine-tuning. This suggests that the proposed features are well matched for the target task.

4.2 Analysis of Coarse Segmentation System

We now examine the value of our PTP framework for predicting whether to "Pull The Plug" on computers and solicit human effort for each image, when segmenting a batch of images with a given budget for human effort/time. Our focus is on initializing segmentation tools. The status quo is typically for humans to create a *coarse object segmentation* input for every image. However, computers also are sometimes employed to automatically position *rectangles* based on the image dimensions (Bernard et al. 2009; Caselles et al. 1997; Chan and Vese 2001). Our system instead intelligently decides which among multiple automatic initialization methods is preferable for each image and then decides whether to involve humans instead (Sect. 3.1.1).

Implementation For each image, our system deploys either (a) the algorithm-generated result from 14 options with the largest *predicted* Jaccard score or (b) a human-drawn segmentation. We leverage cross-dataset predictions (Sect. 4.1) from our top-performing mask-based predictors to estimate the quality of algorithm-drawn segmentations so that our method cannot inadvertently learn and exploit dataset-specific idiosyncrasies.

Baselines We compare our method to the following hybrid human–computer methods for creating coarse segmentation inputs:

- *Perfect predictor* For each image, this system deploys the algorithm-generated result from 14 options that has the largest *actual* Jaccard score. Human involvement is allocated to the images with lowest scores. This predictor reveals the best initializations possible with our system.
- *Chance predictor* For each image, the system randomly deploys one algorithm-generated result from the 14 options. Then, images for human involvement are randomly selected. For a lay person who lacks specialized knowledge of which algorithms work well in their domain, this predictor illustrates the best (s)he can achieve today with the initialization options available in our system.
- *Rectangle* (Bernard et al. 2009; Caselles et al. 1997; Chan and Vese 2001): This method illustrates the commonly-adopted automated method of positioning a bounding rectangle with respect to the image dimensions. Following Chan and Vese (2001), we set the foreground region based on the image boundary. We position the rectangle to occupy the image region after cropping 5% of pixels from the minimum image dimension on all sides. We randomly select images for human involvement.
- *Linear* (Gurari et al. 2016) This is a variant of our approach, as implemented in our prior experiments, that uses a linear regression model instead of the non-linear regression trees. It predicts which of the original eight segmentation options to deploy as input.
- *No-Refinement* This method illustrates the performance of our prediction system in the absence of refinement. Specifically, the system relies on the input crowd-generated and algorithm-generated results as is, rather than the output of the refinement algorithms that modify these inputs.

Experimental design To illustrate the versatility of our initialization system as a general-purpose approach for use with refinement segmentation tools, we integrate our initialization method and the baselines with three tools important in the computer vision and medical imaging communities—GrabCut (Rother et al. 2004), Chan Vese level sets (Chan and Vese 2001), and Lankton level sets (Lankton and Tannenbaum 2008). GrabCut enforces color homogeneity and spatial proximity. The Chan Vese level set method uses global image information to try to separate an image into two homogeneous intensity regions while enforcing smoothness of the object boundary. The Lankton level set method uses local neighborhood statistics for each pixel to separate an object from the background so that there are two homogeneous intensity regions within a band containing the object boundary.

We evaluate each system using all images in Weizmann, IIS, and BU-BIL as well as a random sample of 174 images from MSRA10K. We evaluate with a subset of images from MSRA10K in order to make it comparable in size to the other datasets; 174 is the average number of images in Weizmann (i.e., 100), IIS (i.e., 151), and BU-BIL (i.e., 271). We examine the performance of each initialization method for all 696 images coming from all four datasets as well as for each dataset independently.

For human input, we collect segmentations from crowd workers on Amazon Mechanical Turk. We use the same crowdsourcing system employed in prior work (Jain and Grauman 2013) to collect a coarse segmentation per image.

4.2.1 Fully-Automated Initialization

On average, our system takes 14.51 s to generate all candidate segmentations, 0.52 s to predict which result is the best, and 9.01 s for refinement (i.e., average of 1.33 s for GrabCut, 10.17 s for Chan Vese level set, and 15.53 seconds for Lankton level set).

For each segmentation tool, we compute the average segmentation quality resulting after the tool refines all initializations for all images. For this analysis, the reader should focus on the leftmost points on the plots in Fig. 6 only (i.e., 0% human involvement).

Predicting a best-suited automated input from 14 options produces coarse segmentation estimates that the segmentation tools can refine more successfully than all baselines (i.e., Chance Predictor, Rectangle (Bernard et al. 2009; Caselles et al. 1997; Chan and Vese 2001), Linear (Gurari et al. 2016), No-refinement). For example, the resulting segmen-

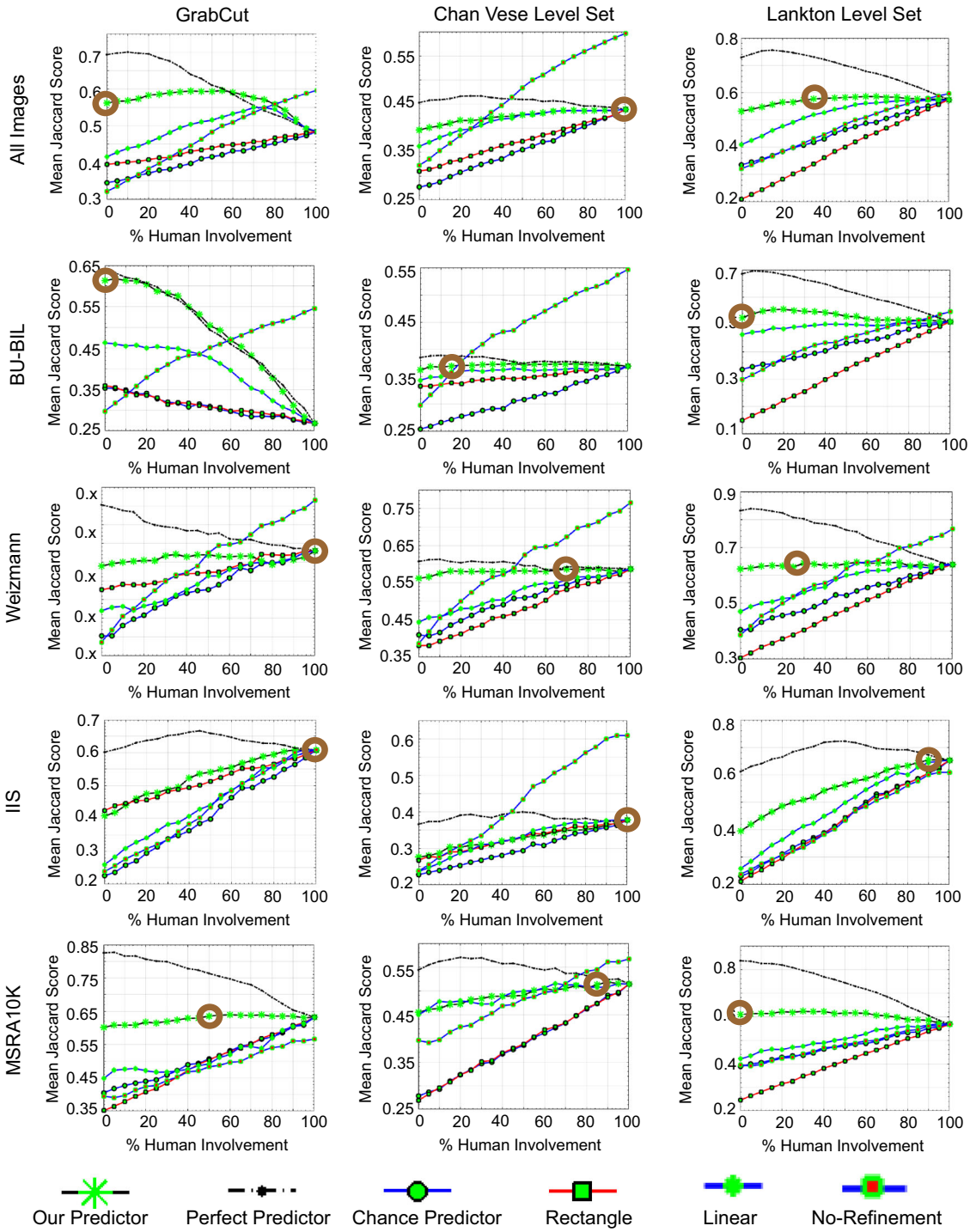


Fig. 6 We compare six methods for distributing varying levels of human involvement to create initializations for three different segmentation refinement tools (cols 1–3) across four datasets (row 1) and each dataset independently (rows 2–5). Each plot shows the mean quality for the segmentations that resulted after the tools refined the initializations. Brown circles identify when our predictor achieves quality comparable to relying exclusively on human input with the least human effort. Compared to relying exclusively on human input, our approach eliminates 55% of human effort with no loss to quality (Color figure online)

tation quality improves by 31% points and 15% points over the Rectangle baseline for the Lankton level set algorithm and GrabCut algorithm respectively (Fig. 6, "All Images"). With respect to our Linear approach (Gurari et al. 2016), we observe the greatest boost from our new approach on the Weizmann dataset; e.g,. 14% points improvement for the

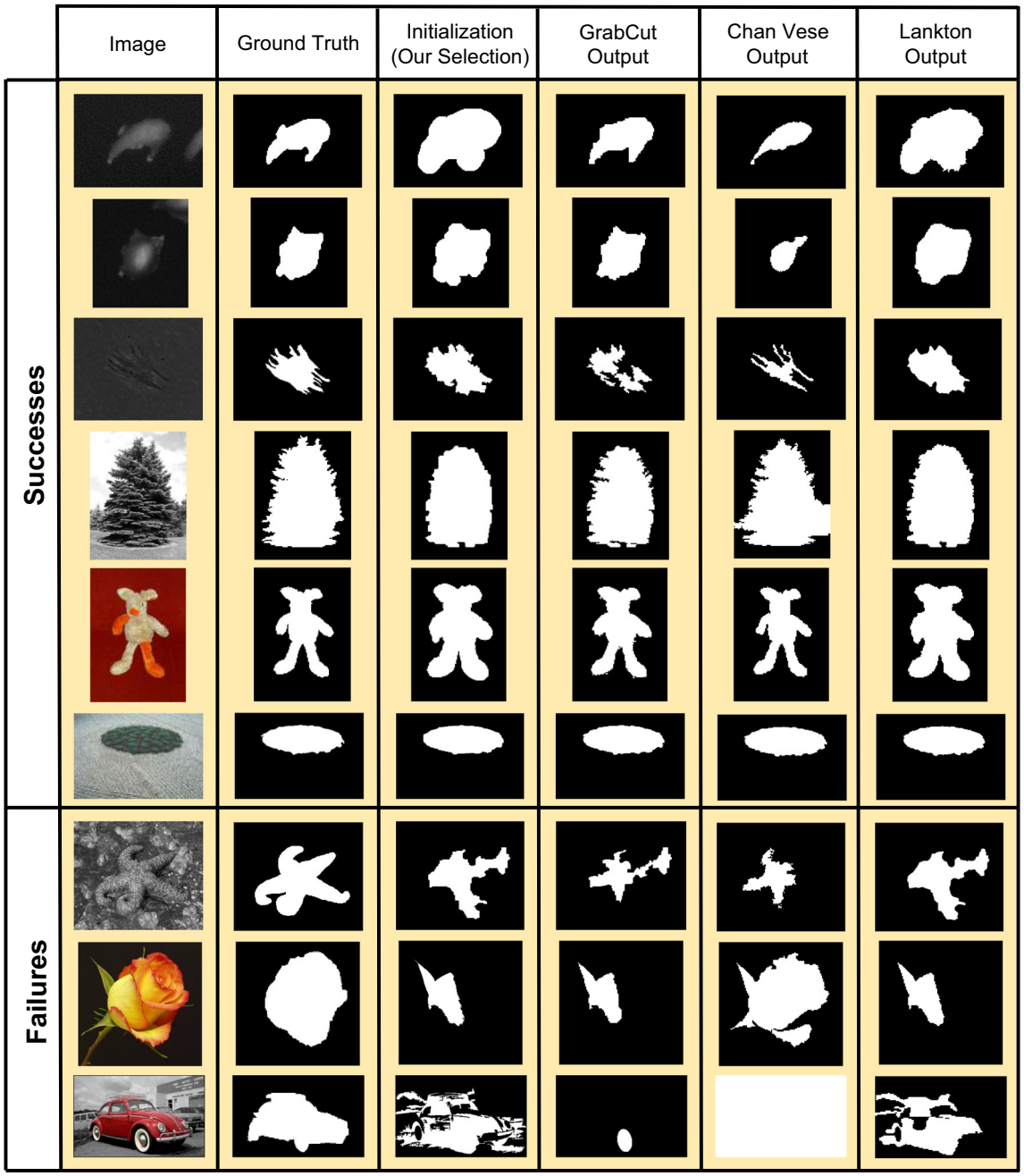


Fig. 7 Illustration of the quality of resulting segmentations created by three segmentation tools from the initial segmentation selected by our system from the 14 initialization options. Sample results are shown for images coming from three imaging modalities: fluorescence microscopy (rows 1–2), phase contrast microscopy (row 3), and everyday images (rows 4–9). These illustrate successes (rows 1–6) and failures (7–9) in creating high quality output segmentations

Lankton level set algorithm. The only exception where our proposed algorithm does not yield better results to the baselines is with the GrabCut algorithm on the IIS dataset; i.e., the Rectangle baseline performs better by approximately 2% points. We hypothesize this exception is because the images typically show more complex scenes and backgrounds, as suggested by the high pixel foreground and background variance in Table 1, which causes initializations to sometimes latch on to a region that does not contain the object of interest. Overall, our findings highlight the value of intelligently predicting a best-suited algorithm per image from multiple options rather than relying on a single initialization approach.

We show qualitative results that illustrate the versatility of our system to initialize the three different segmentation tools in Fig. 7. As shown, given the same initialization, the three segmentation tools can produce very similar segmentations

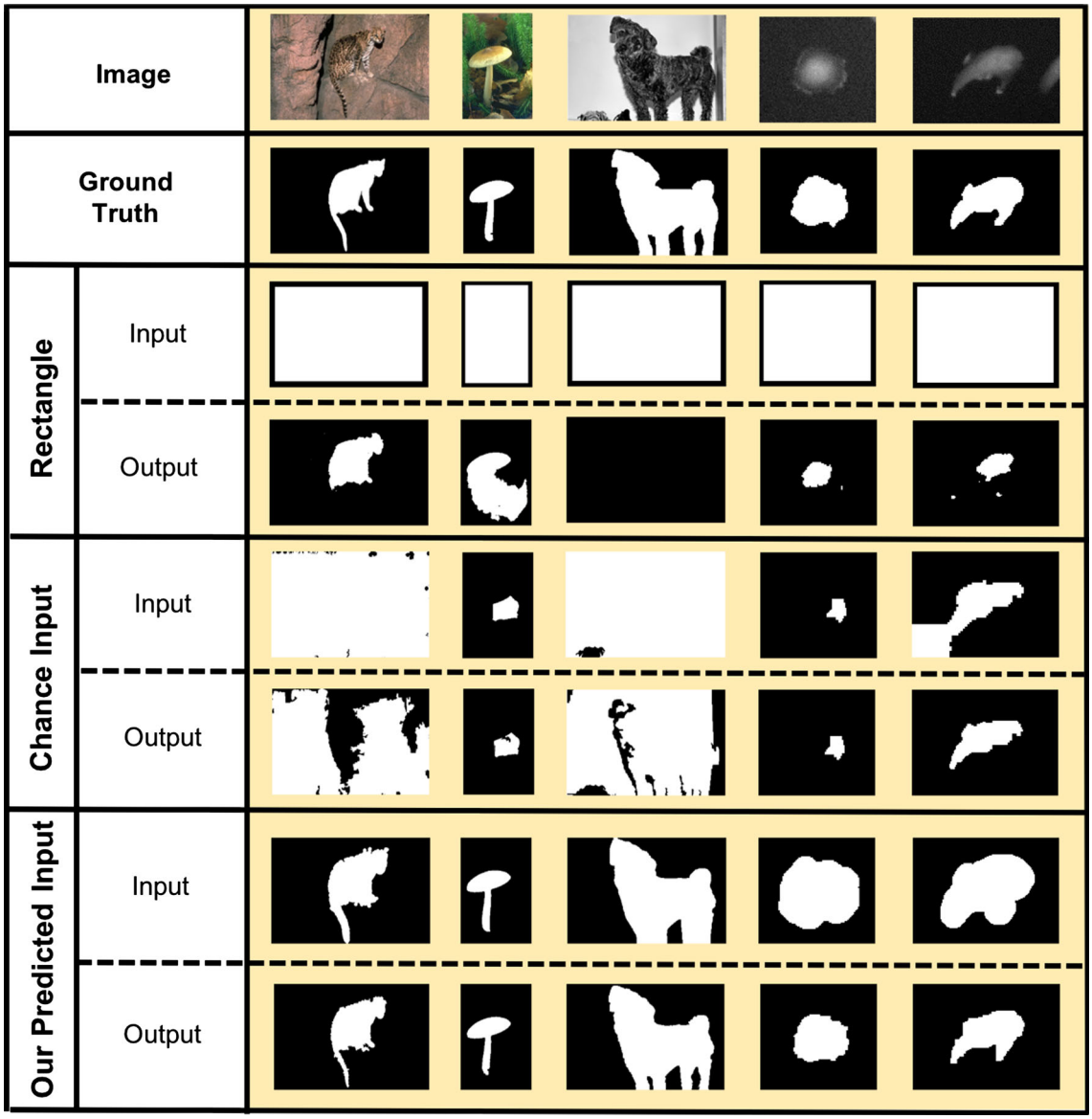


Fig. 8 Performance of the GrabCut algorithm when refining three different initialization approaches ("Input") into final ("Output") segmentations

in some cases (e.g., plot of land in row 6) and dramatically different segmentations in other cases (e.g., spiculated cell shown in row 3). We also observe that a segmentation tool can perform well when using a low quality initialization, as observed for the image of the cell (row 1, GrabCut algorithm). These qualitative results illustrate our quantitative finding that applying a refinement algorithm typically yields considerable improvements over using the input as is ("No-Refinement"); e.g., by over 20% points for the Lankton level set algorithm (Fig. 6, "All Images"). More generally, our system can often automatically produce sufficiently accurate initializations required by segmentation tools to produce segmentations that resemble the ground truth.

We also show qualitative results that illustrate the failure cases of our system (Fig. 7; rows 7–9). In some cases, none of the three segmentation tools perform well when initialized poorly, as observed for the image of the starfish (row 7) and flower (row 8). Additionally, the refinement algorithms can perform poorly in refining an initialization, as observed for the image of the car which has a noisy initialization that roughly segments it out and refinements that do not restore the car (row 9).

We also show results comparing the output from the GrabCut algorithm initialized with our fully-automated segmentation initialization system as well as two baselines in Fig. 8. As observed, the quality of segmentation results is higher with our intelligent selection approach than arbitrar-

ily chosen initial segmentation estimates (i.e., Rectangle, Chance). Still, our approach does not necessarily ensure all fine-grained details of the object boundary are captured, as observed in columns 1 and 4.

Our results highlight that applying the refinement algorithm to the top-predicted coarse segmentation leads to improvements over "No-Refinement" for all three refinement algorithms: GrabCut, Chan Vese level set, and Lankton level set. For example, we observe a 26% improvement when applying the GrabCut algorithm rather than using the coarse segmentations alone (Fig. 6, "All Images"). This demonstrates a benefit of applying refinement algorithms to clean up coarse segmentations.

4.2.2 Reducing Human Initialization Effort

Thus far we have analyzed the impact of our method in a fully unsupervised setting, i.e., 0% human involvement. We next examine the impact of actively allocating human involvement to create *coarse segmentation input* as a function of the budget of human effort available. For each segmentation tool, we compute the average segmentation quality resulting after the tool refines the collection of chosen computer and human initializations for all images. These results are also shown in Fig. 6; i.e., all values greater than 0% human involvement.

Our approach typically outperforms random decisions (i.e., Chance, Rectangle) and our Linear approach (Gurari et al. 2016) regarding how to distribute the initialization effort to humans and computers for all budget levels across all datasets. Our approach also has the potential to outperform all three baselines for all segmentation tools by greater margins given improved prediction accuracy, as exemplified by the Perfect Predictor.

In the more challenging setting of eliminating human effort without compromising segmentation quality, our system yields exciting results. Specifically, our system achieves comparable quality to relying exclusively on crowdsourced input (i.e., 100% human involvement) while using no human involvement for all images for GrabCut and human involvement for 35% of images for Lankton level sets (Fig. 6; see brown circles on figures). Our results reveal that different segmentation tools can tolerate different amounts of unreliable computer input without compromising the overall segmentation quality attained when relying exclusively on human input.

Our findings also highlight what, if any, benefits arise from employing refinement algorithms versus using the input segmentations as is. Overall, we observe that two of the three refinement algorithms yield considerable improvements for most budget levels across all datasets: GrabCut and Lankton level sets. However, we found for these algorithms that it is beneficial to forego refinement when the available human budget can cover 80% or more of the images; i.e., the No-Refinement results tend to outperform Our Predictor. Our findings underscore the importance of selecting a good refinement algorithm and understanding its strengths in order to reap the benefit of our coarse segmentation initialization system.

Given the scalability of crowdsourcing, we employed (non-trained) crowd workers to provide human annotations. However, employing trained experts instead could lead to higher quality results from human initializations and so slightly different quality human-effort trade-offs

4.2.3 Peak Segmentation Quality

Relying on a mix of human and computer efforts can outperform relying on either resource alone to create initial segmentations. For example, peak accuracy for GrabCut with our initialization approach is achieved with 55% human and 45% computer involvement (Fig. 6, GrabCut on "All Images"). There is a 12% point improvement from relying on a mix of human and computer input over human input alone. We attribute this finding to the tool's shrinking bias, which leads GrabCut to perform poorly when the initial boundary does not entirely subsume the true object region. We believe this tendency is especially pronounced for human-drawn course segmentations for the biomedical images, as suggested by the algorithm consistently performing poorly when converting these to final segmentations (Fig. 6, GrabCut on "BU-BIL"; 100% human involvement). In addition, we observe slight performance gains for the Lankton level set algorithm, with the tool fluctuating around a peak plateau value from 35% to 100% human involvement (Fig. 6, Lankton level set on "All Images"). We attribute the latter performance fluctuations to slight differences when the tool expands and shrinks the human and algorithm initializations as needed to recover the desired boundaries. More generally, our findings reveal that intelligently replacing human effort with computer effort can be desirable not only to save money and time, but to also collect higher quality segmentations.

Our findings also demonstrate that the best possible performance across all benchmarked methods is obtained with the Perfect Predictor for two of the three refinement algorithms (GrabCut and Lankton level set) by relying on more computer effort than human effort. As observed, the peak performance arises at $\sim$10% human involvement for both algorithms. A natural question is why does the performance increasingly fall with increasing amounts of human effort. We hypothesize this fall is partially due to the relative lower quality of human-annotated coarse segmentations compared to what is possible with a fully automated approach. Specifically, while the average quality for all human annotations is $\sim$58% (Fig. 6, "No-Refinement" for "All Images"; 100% human involvement), a fully-automated approach yields on

average 71% (i.e., 0% involvement for Perfect Predictor for GrabCut and Lankton level set algorithms). We also hypothesize the performance drop arises partially because of inadequate performance from the refinement algorithms, since refining reasonably high quality coarse segmentations can lead to worse performance than using the coarse segmentations as is for all three refinement algorithms; e.g., ∼60% for "No-Refinement" versus ∼45% for all remaining methods for the GrabCut algorithm (Fig. 6; "No-Refinement" versus all other methods for "All Images"; 100% human involvement).

4.3 Analysis of Fine-Grained Segmentation System

Lastly, we examine the value of our *PTP* framework to predict when to pull the plug on computers and use human annotation instead to create fine-grained segmentations. For this second task, given segmentations from algorithms, the system predicts which images humans should re-annotate in order to recover from failures (Sect. 3.1.2).

Implementation Our system automatically feeds initializations from our fully-automated *Coarse Segmentation* system to the GrabCut algorithm, the top-performing method found in Sect. 4.2.1. Quality estimates of the resulting segmentations are then predicted using our top-performing mask-based predictor (Sect. 4.1). To avoid inadvertently learning and exploiting dataset-specific idiosyncrasies, we again employ the cross-dataset predictors.

Baselines To our knowledge, no prior work addresses predicting when to enlist human versus computer effort for segmentation. We compare our method to the following related methods for creating fine-grained segmentations:
- *Perfect predictor* For each image, the system deploys the initial algorithm-generated result from 14 options that has the largest *actual* Jaccard score, as done in Sect. 4.2. Then, the system ranks the resulting segmentations from the GrabCut algorithm based on the *actual* Jaccard scores. Human involvement is allocated to the images with lowest scores. This predictor reveals the best results possible with our *Fine-Grained Segmentation* system.
- *Chance predictor* For each image, the system deploys the commonly-adopted initialization of positioning a bounding rectangle with respect to the image dimensions (Sect. 4.2, *Rectangle*). Then, images for human involvement are randomly selected. This predictor illustrates the best a user can achieve today.
- *Jain & Grauman* (*J & G*) (Jain and Grauman 2013) This method predicts how to best allocate a given budget of human time to annotate a batch of images. In particular, it predicts whether to have humans draw a segmentation from scratch (54 seconds) versus supply a rectangle (7 seconds) or coarse segmentation (20 seconds) as input to GrabCut. The system was trained on everyday images for GrabCut. We use publicly-available code.

Experimental design To represent images from the three imaging modalities with a similar number of images per modality, we conduct studies on all images from Weizmann, IIS, and BU-BIL. Following prior work (Jain and Grauman 2013), we budget 54 s for each segmentation a human creates from scratch. We examine the impact of actively allocating human effort using a budgeted approach, ranging from no human involvement (0 min) to getting all images from the three datasets manually annotated (470 min). We compute the average segmentation quality resulting for all chosen human-drawn and computer-drawn segmentations at each allotted time budget.

For human input, we collect segmentations from online crowd workers. We measure the quality as the Jaccard similarity of each crowdsourced segmentation to the ground truth.

Results Our system typically outperforms the related state of art *J & G* interactive method (Jain and Grauman 2013) for a wide range of budgets (Fig. 9a). The benefit of our approach is greatest in the range of 0–40% human involvement, typically eliminating 45–100 min of human annotation effort to achieve segmentation quality comparable the *J & G* interactive method (Jain and Grauman 2013). A further advantage of our approach is that, unlike the *J & G* (Jain and Grauman 2013) system, our system works even when human involvement is not available for every image. Specifically, as observed in Fig. 9a, the *J & G* method (Jain and Grauman 2013) only becomes relevant at the budget level that supports human-created bounding boxes for all images (i.e., approximately at 12% human involvement). Our findings highlight the value of our system to save human effort.

Our findings also highlight the importance of a strong predictor for our system. For example, with no human involvement, our proposed approach could improve a further 10% points to achieve the performance of the *Perfect Predictor* (Fig. 9a). Furthermore, our system would yield comparable quality to relying exclusively on crowdsourced workers while eliminating 16% points of human effort, given a perfect predictor (Fig. 9a, *Perfect Predictor*). While there are clear benefits from our approach, a valuable area for future work is to further improve the predictor. For example, while our quality prediction system is currently designed to be agnostic to the refinement algorithm, it could instead be retrained using masks generated by the refinement algorithm towards improving its performance.

Our findings also reveal that relying on a mix of human and computer effort can outperform methods that always assume human involvement. In particular, for the last 55 images assigned to receive human annotations (i.e., images with highest predicted algorithm scores), the system appropriately chooses computer-drawn segmentations over human-drawn segmentations for 16% of images. For those images, com-

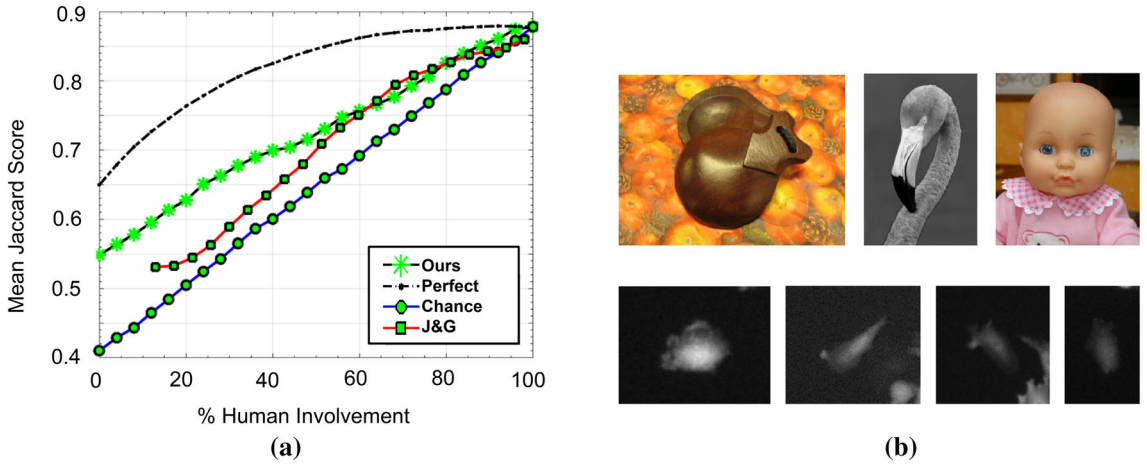


Fig. 9 Predicting when to replace segmentations created by a semi-automatic segmentation tool with segmentations created by (**a**) online crowd workers for 522 images representing three imaging modalities. Our system typically achieves state-of-art accuracy (*J & G* method (Jain and Grauman 2013)) while saving up to 100 min of human effort (i.e., time difference between curves in the human budget range of 0 to 20% human involvement). (**b**) Examples of images which computers segment more similarly to experts than crowd workers. As intended, our system often avoids involving crowd workers for these images

puters create segmentations more similar to the ground truth than crowd workers (i.e., higher Jaccard scores). Example images where algorithms segment better than the crowd are shown in Fig. 9b.

5 Conclusions

We proposed two novel tasks for intelligently distributing segmentation effort between computers and humans. Both tasks rely on our proposed prediction module that predicts the quality of candidate segmentations from three diverse modalities (i.e., visible, phase contrast microscopy, fluorescence microscopy). For the first task of creating initializations that segmentation tools refine, our proposed system eliminated the need for crowdsourced human annotation effort for an average of 55% of images while preserving the resulting segmentation quality achieved when relying exclusively on human input. For the second task of creating high quality segmentation results, our proposed system consistently preserved the resulting segmentation quality from a state of art interactive segmentation tool while regularly eliminating human annotation time. Our work can relieve lay people from requiring domain expertise to identify which segmentation algorithm to use by automatically identifying which from numerous popular algorithms is best. Moreover, it guides end users to direct their limited time to where their efforts will be of most value.

Valuable future research would include generalizing this work by designing a larger-scale system that supports more algorithms and image sets. Towards this aim, next steps include creating a centralized, online repository of segmentation algorithms to which anyone can contribute and identifying the ideal, complementary subset of algorithms to use in order to avoid the computational overhead of applying all segmentation algorithms to each image. Next steps also include generalizing the idea of automatically soliciting human input when algorithms fail for other tasks such spatio-temporal tracking of objects in videos. Key issues to address include how to avoid error drift through the 3D image stack and what amount of 3D context should be presented to humans to support their video annotation efforts.

Acknowledgements The authors thank the anonymous crowd workers for participating in our experiments. This work is supported in part by National Science Foundation funding to DG (IIS-1755593), a gift from Adobe to DG, National Science Foundation funding to MB (IIS-1421943), a Google Faculty Award to MB, AWS Machine Learning Research Award to KG, IBM Faculty Award to KG, IBM Open Collaborative Research Award to KG, and a gift from Qualcomm to KG.

References

Alpert, S., Galun, M., Basri, R., & Brandt, A. (2007). Image segmentation by probabilistic bottom-up aggregation and cue integration. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 1–8).

Arbelaez, P., Maire, M., Fowlkes, C., & Malik, J. (2011). Contour detection and hierarchical image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *33*(5), 898–916.

Arbeláez, P., Pont-Tuset, J., Barron, J., Marques, F., & Malik, J. (2014) Multiscale combinatorial grouping. In *IEEE conference on computer vision and pattern recognition* (pp. 328–335).

Ballard, D. (1981). Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*, *13*(2), 111–122.

Batra, D., Kowdle, A., Parikh, D., Luo, J., & Chen, T. (2010) iCoseg: Interactive co-segmentation with intelligent scribble guidance. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 3169–3176). IEEE.

Bernard, O., Friboulet, D., Thevenaz, P., & Unser, M. (2009). Variational b-spline level-set: A linear filtering approach for fast, deformable model evolution. *IEEE Transactions on Image Processing*, *18*(6), 1179–1191.

Biswas, A., & Parikh, D. (2013) Simultaneous active learning of classifiers & attributes via relative feedback. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 644–651).

Branson, S., Grant, V. H., Wah, C., Perona, P., & Belongie, S. (2014). The ignorant led by the blind: A hybrid human-machine vision system for fine-grained categorization. *International Journal of Computer Vision*, *108*, 3–29.

Carlier, A., Charvillat, V., Salvador, A., i Nieto, X. G., & Marques, O. (2014). Click'n'Cut: Crowdsourced interactive segmentation with object candidates. In *International ACM workshop on crowdsourcing for multimedia* (pp. 53–56).

Carreira, J., Sminchisescu, C. (2010). Constrained parametric min-cuts for automatic object segmentation. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 3241–3248).

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

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

Cheng, M., Mitra, N. J., Huang, X., Torr, P. H. S., & Hu, S. (2014). Global contrast based salient region detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *37*(3), 569–582.

Chittajallu, D. R., Florian, S., Kohler, R. H., Iwamoto, Y., Orth, J. D., Weissleder, R., et al. (2015). In vivo cell-cycle profiling in xenograft tumors by quantitative intravital microscopy. *Nature Methods*, *12*(6), 577–585.

Cui, J., Yang, Q., Wen, F., Wu, Q., Zhang, C., Gool, L. V., & Tang, X. (2008) Transductive object cutout. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 1–8).

Endres, I., & Hoiem, D. (2010). Category independent object proposals. In *European conference on computer vision (ECCV)* (pp. 575–588).

Everingham, M., Gool, L. V., Williams, C. K. I., Winn, J., & Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, *88*(2), 303–338.

Glenn, D. R., Lee, K., Park, H., Weissleder, R., Yacoby, A., Lukin, M. D., et al. (2015). Single-cell magnetic imaging using a quantum diamond microscope. *Nature Methods*, *12*, 736–738.

Grady, L., Jolly, M. P., & Seitz, A. (2011). Segmentation from a box. In *IEEE international conference on computer vision (ICCV)* (pp. 367–374).

Gulshan, V., Rother, C., Criminisi, A., Blake, A., & Zisserman, A. (2010). Geodesic star convexity for interactive image segmentation. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 3129–3136).

Gurari, D., He, K., Xiong, B., Zhang, J., Sameki, M., Jain, S. D., et al. (2018). Predicting foreground object ambiguity and efficiently crowdsourcing the segmentation(s). *International Journal on Computer Vision (IJCV)*, *126*, 714–730.

Gurari, D., Jain, S. D., Betke, M., & Grauman, K. (2016). Pull the plug? predicting if computers or humans should segment images. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 382–391).

Gurari, D., Theriault, D., Sameki, M., & Betke, M. (2014). How to use level set methods to accurately find boundaries of cells in biomedical images? Evaluation of six methods paired with automated and crowdsourced initial contours. In *Conference on medical image computing and computer assisted intervention (MICCAI): Interactive medical image computation (IMIC) workshop* (pp. 9).

Gurari, D., Theriault, D., Sameki, M., Isenberg, B., Pham, T. A., Purwada, A., Solski, P., Walker, M., Zhang, C., Wong, J. Y., & Betke, M. (2015). How to collect segmentations for biomedical images? A benchmark evaluating the performance of experts, crowdsourced non-experts, and algorithms. In *IEEE winter conference on applications in computer vision (WACV)* (pp. 8).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *IEEE conference on computer vision and pattern recognition* (pp. 770–778).

Jain, S. D., & Grauman, K. (2013). Predicting sufficient annotation strength for interactive foreground segmentation. In *IEEE international conference on computer vision (ICCV)* (pp. 1313–1320).

Jain, S. D., Xiong, B., & Grauman, K. (2017). Fusionseg: Learning to combine motion and appearance for fully automatic segmention of generic objects in videos. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)* (Vol. 1).

Kohlberger, T., Singh, V., Alvino, C., Bahlmann, C., & Grady, L. (2012). Evaluating segmentation error without ground truth. In *Medical image computing and computer assisted intervention (MICCAI)* (pp. 528–536).

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In *Advances in neural information processing systems (NIPS)* (pp. 1097–1105).

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

Lempitsky, V., Kohli, P., Rother, C., Sharp, T. (2009). Image segmentation with a bounding box prior. In *IEEE international conference on computer vision (ICCV)* (pp. 277–284).

Li, C., Kao, C. Y., Gore, J. C., & Ding, Z. (2008). Minimization of region-scalable fitting energy for image segmentation. *IEEE Transactions on Image Processing*, *17*(10), 1940–1949.

Li, H., Meng, F., Luo, B., & Zhu, S. (2014). Repairing bad co-segmentation using its quality evaluation and segment propagation. *IEEE Transactions on Image Processing*, *23*(8), 3545–3559.

Liu, D., Xiong, Y., Pulli, K., & Shapiro, L. (2011). Estimating image segmentation difficulty. In *Machine learning and data mining in pattern recognition* (pp. 484–495).

Liu, T., Yuan, Z., Sun, J., Wang, J., Zheng, N., Tang, X., et al. (2011). Learning to detect a salient object. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *33*(2), 353–367.

Maitra, M., Gupta, R. K., & Mukherjee, M. (2012). Detection and counting of red blood cells in blood cell images using Hough transform. *International Journal of Computer Applications*, *53*(16), 18–22.

Martin, D., Fowlkes, C., Tal, D., & Malik, J. (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *International conference on computer vision (ICCV)* (Vol. 2, pp. 416–423).

Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, *9*(1), 62–66.

Rother, C., Kolmogorov, V., & Blake, A. (2004). GrabCut: Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, *3*, 309–314.

Settles, B. (2010). Active learning literature survey. Technical report, University of Wisconsin, Madison.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint* arXiv:1409.1556.

Vijayanarasimhan, S., & Grauman, K. (2011). Cost-sensitive active visual category learning. *International Journal of Computer Vision*, *91*, 24–44.

Wah, C., Maji, S., & Belongie, S. (2015). Learning localized perceptual similarity metrics for interactive categorization. In *IEEE Winter*

conference on applications in computer vision (WACV) (pp. 502–509).

Wu, J., Zhao, Y., Zhu, J., Luo, S., & Tu, Z. (2014). MILCut: A sweeping line multiple instance learning paradigm for interactive image segmentation. In *IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 256–263).