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

Danna Gurari Suyog Dutt Jain Margrit Betke Kristen Grauman

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 proposed prediction module that estimates the quality of given algorithm-drawn segmentations. We demonstrate the value of the framework for two novel tasks related to "pulling the plug" on computer and human annotators. Specifically, we implement two systems that automatically decide, for a batch of images, when to replace 1) humans with computers to create coarse segmentations required to initialize segmentation tools and 2) computers with humans to create 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 three diverse datasets representing visible, phase contrast microscopy, and fluorescence microscopy images.

1. Introduction

A common question people ask when needing to annotate images is whether automated options are sufficient for their images 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 only available for $K\%$ of images.

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

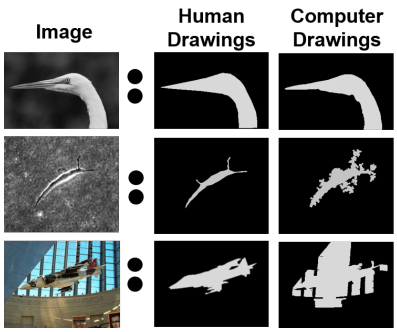

Figure 1. Use a human-drawn or computer-drawn segmentation? We propose a task of automatically deciding when to "pull the plug" on human annotators and use computers instead to create the initial foreground segmentations (rows 1, 2) that segmentation tools refine. We also propose a task of automatically deciding when to "pull the plug" on computers (row 3) and use humans instead to create high quality segmentations.

(**Figure 1**). This is because algorithms embed assumptions about how to separate an object from the background that are relevant for specific object and background appearances, yet restrict their widespread applicability [4, 12, 26, 34, 35]. 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 *consistently* high quality segmentations currently face a brute force approach of reviewing all images with available algorithm-drawn segmentations to identify images that should be re-annotated by humans.

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 [9, 18, 20, 23, 27, 35, 39]. 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 [23]. A weakness of relying on humans is that for numerous methods, including level set based methods [6, 12, 26, 28], humans typically have to wait

for minutes or more per image to validate whether the tool successfully converts their coarse input to high quality segmentations. Intuitively, one may expect that computers at times can create good enough segmentations to replace human initialization effort (e.g., **Figure 1**, rows 1 & 2) 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.

To the best of our knowledge, this work is the first to predict when to "pull the plug" on humans or computers for segmenting images. We address two novel tasks. First, we propose a system that intelligently allocates computer effort to replace human effort 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 automated methods segmented poorly. Both systems are designed to empower users to consistently collect higher quality object segmentations with segmentation tools while using considerably less human involvement. More broadly, our systems could be exploited to efficiently create segmentations as input for downstream tasks (e.g., object recognition, tracking).

Interactive *co-segmentation* methods address the issue of relying on human input to initialize segmentation tools for every image in a batch [5, 14, 29]. 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 [12, 26] 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 [36]. Specifically, active learners try to identify the most impactful, yet least expensive information necessary to train accurate prediction models [7, 36, 37]. For example, some methods iteratively supplement a training dataset with images predicted to require little human annotation time to label [37]. Other methods actively solicit human feedback to identify features with stronger predictive power than those currently available [7]. 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 [3, 10, 15, 24]. 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 [15]. Another method proposes an absolute segmentation difficulty measure based on the image content alone [30]. However, this method does not account for differences in segmentation tools and that they perform differently when applied to segment 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 [10, 24]. In particular, we also perform supervised learning to train a regression model. Unlike prior work, which was proposed independently in the medical [24] and computer vision [10] communities, we aim to develop a single prediction model that is applicable across domains. Consequently, we populate our training data with segmentations resulting from a variety of algorithms on images from three imaging modalities (visible, phase contrast microscopy, fluorescence microscopy). Our approach consistently predicts well, outperforming a widely-used method [10], on three diverse datasets.

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 outperform relying on humans or computers alone. For example, hybrid systems combine non-expert and algorithm strengths to perform the challenging fine-grained bird classification task typically performed by experts [8, 38]. While our hybrid system design complements existing work by also demonstrating the advantages of combining human and computer efforts, our work differs by addressing the image segmentation task rather than the class labeling task.

2. Segmentations by Humans or Computers?

We first describe two prediction systems for creating different levels of segmentations detail (**Section 2.1**). Then, we describe the module used by both systems to predict the quality of algorithm-generated segmentations (**Section 2.2**).

2.1. Batch Allocation of Humans & Computers

We call our resource allocation framework *PTP* which reflects that the system, for each image in a batch, predicts whether to "**P**ull **T**he **P**lug" on humans or computers. In other words, our framework involves predicting for each image whether the annotation should come from a human or computer. We implement two *PTP* systems to create 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.

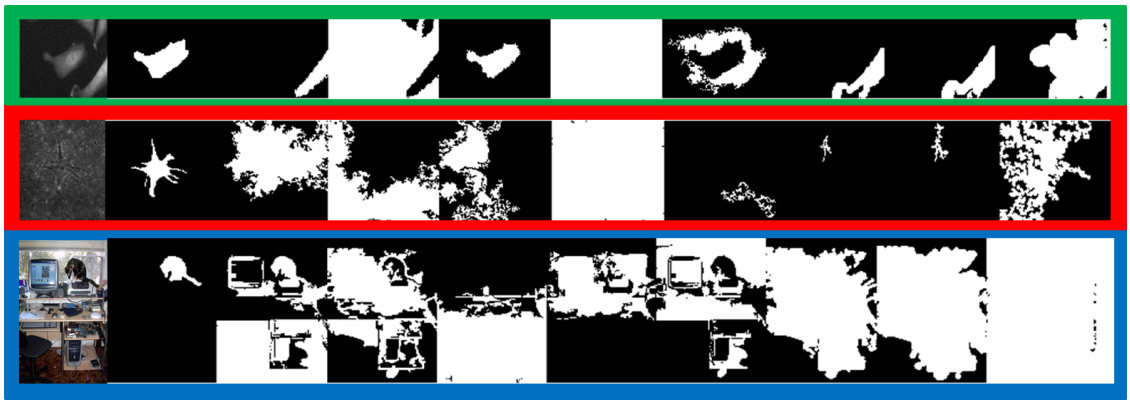

Figure 2. We propose a system to predict when to delegate the task of creating coarse segmentations to an algorithm or a human. The system decides based on a predicted similarity of each algorithm-generated segmentation (i.e., last eight segmentations per row) to the unobserved ground truth (i.e., first segmentation per row). Our system is designed for use across domains, to demarcate the foreground object in fluorescence microscopy (row 1), phase contrast microscopy (row 2), and everyday (row 3) images.

Like existing interactive segmentation methods, we assume the user is interested in a primary foreground object [9, 18, 27, 35, 39]. 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 [2, 16, 33].

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 [5, 14, 29]. The motivation of the system design is to remain agnostic to the particular segmentation tool. Since some segmentation tools require minutes or more to refine a single initialization, we limit our system to run a segmentation tool exactly once per image with one input. Consequently, in the interest of increasing the chance of computer success, our system deploys the best predicted algorithm from a larger list of eight options for each image.

This system involves six key steps to segment a given batch of images. First, eight algorithm-drawn foreground segmentations[1] are collected per image (**Figure 2**). Our motivation is to employ fully-automated algorithms applicable across the image modalities investigated in this paper (visible, phase contrast microscopy, fluorescence microscopy). Then, for each image, the quality of each candidate segmentation is predicted using our proposed prediction system discussed in Section 2.2. Third, the top-scoring segmentation per image is selected as the computer choice. Next, all images are sorted based on the selected computer choices, from highest to lowest predicted quality scores. Fifth, the system allocates the available human budget to create coarse segmentations for the allotted number of images with the lowest predicted quality scores. Finally, all coarse segmentations created by humans and computers are fed to the segmentation tool of interest for refinement.

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 previous system elicits coarse human input to initialize a segmentation tool, we now propose a 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. Then, each coarse segmentation is refined by a segmentation tool. Next the prediction framework is applied to all resulting segmentations from the segmentation tool to estimate the quality of each

[1] The system applies algorithms used in current literature for foreground segmentation [13, 17, 32]: Otsu thresholding[34], adaptive thresholding, and Hough Transform with circles [4] . The system applies Otsu thresholding and its complement. The system also applies adaptive thresholding using the local median from a window size of 45 pixels and its complement as well as a third variant using the local mean from a window size of 45 pixels. Finally, the system applies three variants of Hough Transforms using a circle radius of 3, 5, and 10. Our system then post-processes each binary mask by filling all holes and keeping only the largest object.

While other algorithms could easily be integrated into our system, we found our choices create similar quality for initial segmentations. Specifically, across the three datasets in our experiments, our choices yield an average quality (Jaccard index) of 0.59 using the best option per image compared to 0.57 using MCG's best option from 8 top-ranked candidates [3], 0.59 using CPMC's best option from 8 top-ranked candidates [10], and 0.17 using [31].

result. 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.

2.2. Predicting Segmentation Quality

Embedded in both the *Coarse* and *Fine-Grained* segmentation systems is a module which automatically predicts the similarity of a given segmentation to an unseen ground truth segmentation. We propose as our prediction framework a regression model in order to capture that algorithm-drawn segmentations can range in quality from complete failures to nearly perfect (**Figures 1, 2**). Our key design decisions lie in how to generate training data and choose predictive features.

Training Instances. We aim to populate our training data with segmentation masks that reflect the transition of segmentation quality from perfect (i.e., ground truth), to reasonable human mistakes, to a variety of failure behaviors. Towards this goal, our system collects 11 binary segmentation masks per training image.

We first derive a variety of binary masks using the same fully-automated algorithms leveraged in our *Coarse* segmentation system. Specifically, our system produces eight segmentations per training image using multiple implementations of the algorithms Hough Transform with Circles [4], Otsu Thresholding [34], and adaptive thresholding. An important distinction of our chosen segmentation algorithms compared to alternative tools [12, 35] is that they do not incorporate regularizer terms that can conceal typical failure behaviors, e.g., smoothing highly-jagged edges. Consequently, the different algorithms capture a variety of types of failure behaviors (**Figure 2**).

Given that the training data may be insufficiently populated with higher-scoring segmentations (if all eight algorithm implementations consistently fail), our system augments three binary masks based on the ground truth segmentations. The system uses the ground truth directly. 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.

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

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 fail big when they fail, manifesting appearances unlike what one would expect from widely meaningful object shapes (**Figure 2**). We propose nine features derived from the binary segmentation mask to capture the failure behaviors. We hypothesize that, in aggregation, these features may account for objects of different shapes and sizes. In results, we will examine their advantages over an off-the-shelf state of the art image descriptor, i.e., based on CNNs.

Segmentation Boundary. When algorithms fail, resulting segmentations often have boundaries characterized by an 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 implement 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. 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*.

See Section 3 for an analysis of the variability of these cues measured for objects observed within diverse datasets.

Regression Model. We train a multiple linear regression model with the aforementioned training data. This model leads to easy to interpret, intuitive systems as it indicates how to predict the segmentation quality from a weighted combination of predictive features. Formally, the model is represented as $\mathrm{y} = X\boldsymbol{\beta} + \boldsymbol{e}$ where $\boldsymbol{y}$ denotes an n-

dimensional vector of segmentation quality scores, X denotes a matrix containing feature vectors that characterize every training instance, $\boldsymbol{\beta}$ denotes the model parameters to be learned, and $\boldsymbol{e}$ denotes errors measured between actual quality scores (y) and predicted quality scores ($X\beta$). The objective is to learn $\boldsymbol{\beta}$ so that e is minimized. We train models with WEKA [22] using M5 feature selection.

3. Experiments and Results

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

Datasets. We evaluate our methods on three datasets that represent three imaging modalities: Boston University Biomedical Image Library (BU-BIL:1-5) [21] includes 271 gray-scale images coming from three fluorescence microscopy image sets and two phase contrast microscopy image sets, Weizmann [1] consists of 100 grayscale images showing a variety of everyday objects, and Interactive Image Segmentation [19] (IIS) includes 151 RGB images showing a variety of everyday objects. Each dataset includes human-drawn segmentations that serve as pixel-accurate ground truth segmentations for evaluation.

Together, the three datasets exhibit large variability with respect to object and image properties (**Table 1**). The datasets depict objects that vary greatly in size (e.g., BU-BIL vs IIS), coverage of the image (e.g., BU-BIL vs Weizmann), shape (i.e., large Shape $\boldsymbol{\sigma}$ for all datasets), and texture (i.e., large FG Var $\boldsymbol{\sigma}$ for all datasets). Furthermore, our analysis suggests that image backgrounds can be complicated and/or cluttered (i.e., large BG Var $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$). This diversity is important to ensure our method is challenged to learn generic cues predictive of segmentation failure.

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 (Sec. 2.2; 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	
	$\boldsymbol{\mu}$	$\boldsymbol{\sigma}$	$\boldsymbol{\mu}$	$\boldsymbol{\sigma}$	$\boldsymbol{\mu}$	$\boldsymbol{\sigma}$
Area	7927	13,109	24,315	16,815	40,119	41,387
X Loc	126	129	146	29	251	80
Y Loc	115	106	158	61	223	63
Shape	0.48	0.25	0.41	0.2	0.4	0.2
$\frac{\text{FG Area}}{\text{Image Area}}$	0.12	0.04	0.27	0.14	0.19	0.12
FG Var	54	51	1663	1271	2227	1909
BG Var	28	36	540	835	1568	1521

3.1. Quality Prediction for Algorithm Set

We first analyze the predictive power of our proposed framework (**Section 2.2**) to automatically estimate the quality of foreground object segmentations.

Baselines. We compare our method to the *CPMC [10]* approach that also predicts a Jaccard score indicating the quality of a given object 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.

Given the recent rise of CNN features as standard baselines for learning, we also examine the value of a CNN baseline for making predictions. We employ the same training instances using features extracted from the last fully connected layer of AlexNet [25] to train linear regression models. Consequently, each training instance is characterized with a 4096-dimensional vector that is extracted from the image patch created by using the bounding box of the automatically generated segmentation.

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-Set Generalization. To minimize concerns that prediction successes are due to over-fitting to the statistics of a particular dataset, we first evaluate how well our prediction models trained on two of the datasets perform on the third dataset. Overall, our approach performs well, as indicated by high CCs and low MAEs (**Table 2**, row 3). The system is successful, even when trained on completely disjoint datasets; e.g., what the system learned on everyday images (Weizmann, IIS) can successfully be leveraged on biomedical images (BU-BIL: CC = 0.61). 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.

While the CPMC method was designed to generalize across different object types, it had less predictive strength than our approach on all studied datasets (**Table 2**, row

Table 2. Comparison of our model with CPMC [10] and CNN features [25] for predicting the Jaccard score indicating the quality of a foreground segmentation. We report performance scores for our method learned with cross-set training ("Ours:C") as well as single-set training ("Ours:S"). Higher correlation coefficient (CC) scores and lower mean absolute error (M) scores are better.

	BU-BIL		Weizmann		IIS		All	
	CC	M	CC	M	CC	M	CC	M
[10]:C	0.36	0.33	0.61	0.32	0.67	0.31	0.53	0.32
CNN:C	-0.01	3.22	-0.1	26.7	-0.01	45	NA	NA
Ours:C	0.61	0.31	0.64	0.24	0.68	0.22	NA	NA
Ours:S	**0.69**	**0.18**	**0.69**	**0.2**	**0.78**	**0.18**	**0.68**	**0.2**

1 versus row 3). This suggests a possible value in learning the statistics of specific tools one intends to use rather than relying on one-size-fits-all approaches. In addition, CPMC's greater error on the everyday images (Weizmann & IIS; MAE scores) highlights a potential value of populating training data with images from different modalities to promote learning generic algorithm behavior rather than particular data properties. Finally, our clear predictive strength over CPMC on the biomedical images (BU-BIL: CC scores of 0.36 vs 0.61) reveals a plausible limitation that intensity features do not generalize well for objects observed in images captured with different image acquisition technologies, while our binary mask features remain relevant across domains.

We observe that the off-the-shelf CNN feature yields negligible predictive power (**Table 2**, row 2). We hypothesize the high MAE arises from an accumulation of errors due to using a high dimensional feature space. Our results further support our findings that the characteristics of segmentation errors are robustly and sufficiently learned from a small set of features describing the binary mask alone.

Ours: Single-Set Analysis. We next evaluate our prediction framework per dataset (i.e., Weizmann, IIS, BU-BIL) as well as across the three datasets (All). To evaluate, we train and test each of the four configurations using 10-fold cross-validation. We consistently observe performance gains over CPMC and cross-set results (**Table 2**, row 4 versus rows 1–3). These findings highlight a possible benefit of learning how an algorithm behaves with a particular type of image set, when one can know the image type to be encountered at test time.

3.2. Initializing Segmentation Tools

We next examine the value of our *PTP* framework to predict when to pull the plug on human annotators and use computers instead, when segmenting a batch of images with a given human budget. Our focus is on initializing segmentation tools. The status quo is either that humans create *coarse object segmentation* input for every image or computers automatically position *rectangles* based on the image dimensions [6, 11, 12]. Our system, instead, intelligently decides which among multiple automatic initialization methods is preferable for each image and then decides whether to involve humans instead (Section 2.1, *Coarse Segmentation* system).

We evaluate with all 522 images from Weizmann, IIS, and BU-BIL. We collect a coarse segmentation per image from crowd workers on Amazon Mechanical Turk. We compare the following methods for creating coarse segmentation inputs:

- *Ours:* For each image, the system deploys either a) the algorithm from eight options that has the largest *predicted* Jaccard score or b) a human. We leverage cross-dataset predictions (Section 3.1) to estimate the quality of algorithm-generated segmentations. We chose this predictor so our method cannot inadvertently learn and exploit any dataset-specific idiosyncrasies.

- *Perfect Predictor:* For each image, this system deploys the algorithm from eight options that has the largest *actual* Jaccard score. Images are then ordered by the actual quality scores. Human involvement is allocated to the images with lowest quality scores. This predictor reveals the best initializations possible with our system.

- *Chance Predictor:* For each image, the system randomly deploys one algorithm from the eight options. Then, images for human involvement are randomly selected. This predictor illustrates the best a user can achieve today with the initialization options available in our system.

- *Rectangle [6, 11, 12]:* This method illustrates the commonly-adopted automated method of positioning a bounding rectangle with respect to the image dimensions. Following [12], 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.

To illustrate the versatility of our initialization system as a general-purpose approach for use with segmentation tools, we integrate our initialization method and the baselines with three tools important in the computer vision and medical imaging communities - Grab Cut [35], Chan Vese level sets [12], and Lankton level sets [26] (**Figure 4**).

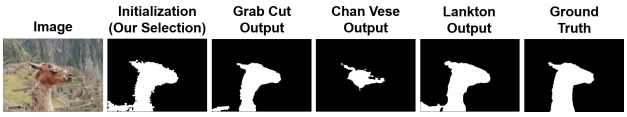

Figure 4. Illustration of the quality of resulting segmentations created by three segmentation tools from the initial segmentation selected by our system from the eight initialization options.

Fully-Automated Initialization. For each segmentation tool, we compute the average segmentation quality resulting after the tool refines all computer-generated initializations for all 522 images. As seen on the left side of the three plots (**Figure 3**, 0% human involvement), predicting a best-suited automated input from eight options produces coarse segmentation estimates that the segmentation tools can refine more successfully than existing baselines (i.e., Chance Predictor, Rectangle). For example, for the Lankton level set algorithm, the resulting segmentation quality improves by 20 percentage points over the Rectangle baseline by using our approach. The one exception is with Grab Cut initialized with the Rectangle baseline. We hypothesize this exception is due to Grab Cut's shrinking bias, which means Grab Cut cannot recover when the initialization occupies a

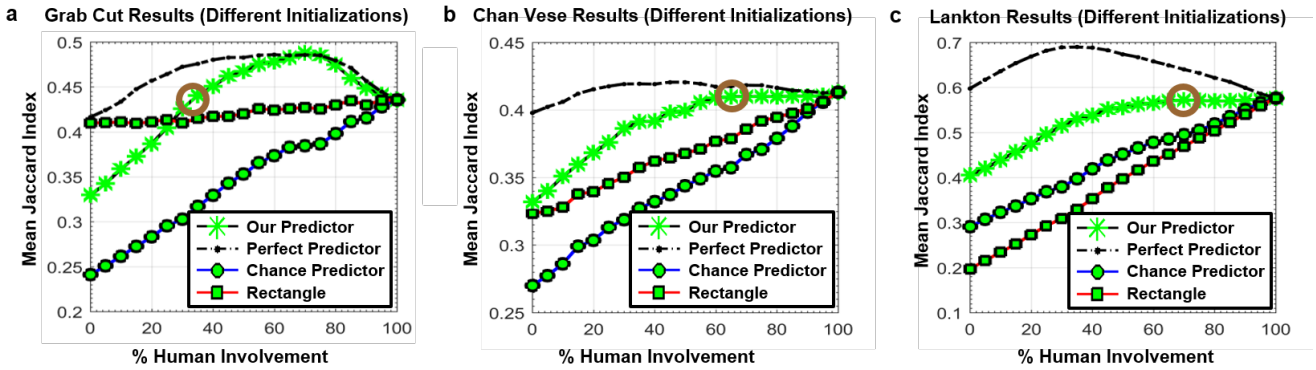

Figure 3. We compare four methods for distributing varying levels of human involvement to create initializations for three segmentation tools (**a-c**). Each plot shows the mean quality for 522 segmentations that resulted after the tools refined the initializations. Our predictor, which identifies the best input option produced by eight algorithms and a human, facilitates segmentation quality comparable to today's status quo (Rectangle, Chance Predictor) with significantly less human involvement. The brown circles identify where our system achieves comparable segmentation quality to relying exclusively on human input. On average, our approach eliminates the need for human annotation effort for 44% of images while achieving segmentation quality comparable to relying exclusively on human input.

region smaller than the object itself.

Reducing Human Initialization Effort. 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 522 images (**Figure 3**). Our approach typically outperforms random decisions (i.e., Chance Predictor, Rectangle) regarding how to distribute the initialization effort to humans and computers for all budget levels. 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 human input (i.e., 100% human involvement) while using computer involvement for 67.5% of images for Grab Cuts, 35% of images for Chan Vese level sets, and 30% of images for Lankton level sets (**Figure 3**; see brown circles). 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.

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 Grab Cuts with our initialization approach is achieved with 70% human and 30% computer involvement (**Figure 3a**). There is a six percentage point improvement from relying on a mix of human and computer input over human input alone. For Chan Vese and Lankton level sets algorithms, performance gains are slight with the tools fluctuating around a peak plateau value from 65% to 100% human involvement (**Figures 3b,c**). We attribute the latter performance fluctuations to slight differences when the two tools expand and shrink the human and algorithm initializations as needed to recover the desired boundaries. We attribute the larger performance gains for Grab Cut to the tool's shrinking bias, which means Grab Cut fails when humans produce boundaries that do not entirely subsume the true object region. More generally, our findings reveal that intelligently replacing human effort with computer effort is not only desirable to save money and time, but also to collect higher quality segmentations.

3.3. Segmentation Tool Output

Lastly, we examine the value of our *PTP* framework to predict when to pull the plug on computers and use human annotation instead. For this second task, given segmentations from algorithms, the system predicts which images humans should re-annotate in order to recover from failures (Section 2.1, *Fine-Grained Segmentation* system).

Implementation. The system automatically feeds initializations from the best stand-alone method (i.e., Hough Transforms with radius 5) to the top-performing Lankton level set algorithm. Quality estimates of resulting segmentations are then predicted using our cross-dataset predictor (Section 3.1).

Baselines. To our knowledge, no prior work addressed predicting when to enlist human versus computer segmentation effort. Therefore, we use as a baseline the related state-of-art system of *Jain & Grauman* [23] (*J & G*) which predicts how to best allocate a given budget of human time to annotate a batch of images. In particular, it predicts whether

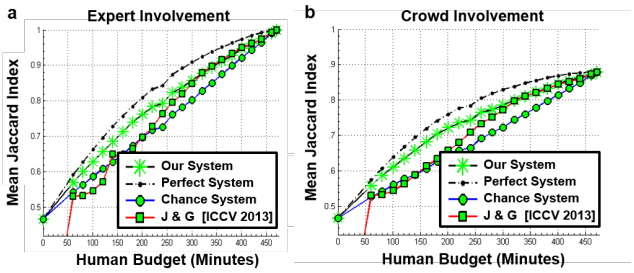

Figure 5. Predicting when to replace segmentations created by a semi-automatic segmentation tool with segmentations created by (**a**) experts and (**b**) online crowd workers for 522 images. With both experts and crowd workers, our system typically achieves state-of-art performance (*J & G* method [23]) while saving up to 60 minutes of human effort (**b**; time difference between curves in the human budget range of 140 to 190 minutes).

to have humans draw a segmentation from scratch (54 seconds) versus supply a bounding box (7 seconds) or coarse segmentation (20 seconds) as input to Grab Cut. The system was trained on everyday images for Grab Cut. We use publicly-available code. Note that the *J & G* [23] system requires human involvement for every image and so only becomes relevant at the budget level that supports human-created bounding boxes for all images (i.e., 61 minutes). Moreover, that system is designed for Grab Cut, whereas our system is agnostic to the segmentation tool.

We also compare the quality of predictions from our approach to perfect and chance predictions for deciding when humans versus computers should segment images.

Experiments. We conduct studies on all 522 images from Weizmann, IIS, and BU-BIL. Following prior work [23], we budget 54 seconds for each segmentation a human creates from scratch. We examine the impact of actively allocating human effort using a budgeted approach, in terms of minutes, ranging from no human involvement (0 minutes) to getting all 522 images manually annotated (470 minutes). 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 analyze both the settings where segmentations are created locally and remotely. For the local setting, we leverage the ground truth segmentations as perfect expert annotations (i.e., Jaccard score of 1). For the web-based setting, we collect segmentations from online crowd workers and measure quality as the Jaccard similarity of each crowdsourced segmentation to the ground truth.

Results. Our system consistently outperforms the baselines for a wide range of budgets, both for expert (**Figure 5a**) and crowd (**Figure 5b**) involvement. For example, the benefit of our approach is greatest at about 50% human budget (i.e., 222 minutes), eliminating an average of 70 minutes of human annotation effort to achieve comparable segmentation quality to the Chance baseline. In addition, our system achieves segmentation quality comparable to the state of art interactive approach [23] but often requires 30-60 minutes less human annotation time. This time savings to achieve same segmentation quality is typically observed in the human budget range of 50 to 220 minutes (**Figure 5a**). Our findings highlight the value of our generic prediction framework today as well as its rich potential for use with future improved segmentation tools.

Figure 6. 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.

Finally, our findings reveal that relying on a mix of human and computer effort can outperform methods that always assume human involvement. In particular, for the last 100 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 10% of images. In other words, for those 10% of images, computers 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 **Figure 6**.

4. Conclusions

We proposed two novel tasks for intelligently distributing segmentation effort between computers and humans. Both tasks relied on our proposed prediction module that successfully predicts the quality of candidate segmentations from three diverse datasets, with stronger predictive capabilities than the baselines. For the first task of creating initializations that segmentation tools refine, our proposed system eliminated the need for human annotation effort for an average of 44% 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 30-60 minutes of human annotation time. We share our code to support application and future extensions of this work (`http://vision.cs.utexas.edu/HybridAlgorithmCrowdSystems/PullThePlug`).

Acknowledgments

The authors gratefully acknowledge funding from the Office of Naval Research (ONR YIP N00014-12-1-0754) and National Science Foundation (IIS-1421943). We thank Mehrnoosh Sameki and Bo Xiong for their assistance with experiments as well as Qinxun Bai, Ajjen Joshi, and the anonymous reviewers for feedback to improve the article.

References

[1] S. Alpert, M. Galun, R. Basri, and A. Brandt. Image segmentation by probabilistic bottom-up aggregation and cue integration. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007. 5

[2] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5):898–916, 2011. 3

[3] P. Arbeláez, J. Pont-Tuset, J. Barron, F. Marques, and J. Malik. Multiscale combinatorial grouping. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 328–335, 2014. 2, 3

[4] D. Ballard. Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111–122, 1981. 1, 3, 4

[5] D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen. iCoseg: Interactive co-segmentation with intelligent scribble guidance. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3169–3176. IEEE, 2010. 2, 3

[6] O. Bernard, D. Friboulet, P. Thevenaz, and M. Unser. Variational b-spline level-set: A linear filtering approach for fast, deformable model evolution. *IEEE Transactions on Image Processing*, 18(6):1179–1191, 2009. 1, 6

[7] A. Biswas and D. Parikh. Simultaneous active learning of classifiers & attributes via relative feedback. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 644–651, 2013. 2

[8] S. Branson, V. H. Grant, C. Wah, P. Perona, and S. Belongie. The ignorant led by the blind: A hybrid human–machine vision system for fine-grained categorization. In *International Journal of Computer Vision*, volume 108, pages 3–29, 2014. 2

[9] A. Carlier, V. Charvillat, A. Salvador, X. G. i Nieto, and O. Marques. Click'n'Cut: Crowdsourced interactive segmentation with object candidates. In *International ACM Workshop on Crowdsourcing for Multimedia*, pages 53–56, 2014. 1, 3

[10] J. Carreira and C. Sminchisescu. Constrained parametric min-cuts for automatic object segmentation. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3241–3248, 2010. 2, 3, 5

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

[12] T. Chan and L. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–277, 2001. 1, 2, 4, 6

[13] D. R. Chittajallu, S. Florian, R. H. Kohler, Y. Iwamoto, J. D. Orth, R. Weissleder, G. Danuser, and T. J. Mitchison. In vivo cell-cycle profiling in xenograft tumors by quantitative intravital microscopy. *Nature Methods*, 12(6):577–585, 2015. 3

[14] J. Cui, Q. Yang, F. Wen, Q. Wu, C. Zhang, L. V. Gool, and X. Tang. Transductive object cutout. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008. 2, 3

[15] I. Endres and D. Hoiem. Category independent object proposals. In *European Conference on Computer Vision (ECCV)*, pages 575–588, 2010. 2

[16] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010. 3

[17] D. R. Glenn, K. Lee, H. Park, R. Weissleder, A. Yacoby, M. D. Lukin, H. Lee, R. L. Walsworth, and C. B. Connolly. Single-cell magnetic imaging using a quantum diamond microscope. *Nature Methods*, pages 736–738, 2015. 3

[18] L. Grady, M. P. Jolly, and A. Seitz. Segmentation from a box. In *IEEE International Conference on Computer Vision (ICCV)*, pages 367–374, 2011. 1, 3

[19] V. Gulshan, C. Rother, A. Criminisi, A. Blake, and A. Zisserman. Geodesic star convexity for interactive image segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3129–3136, 2010. 5

[20] D. Gurari, D. Theriault, M. Sameki, and M. Betke. 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. *Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI): Interactive Medical Image Computation (IMIC) Workshop*, page 9 pp., 2014. 1

[21] 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. 5

[22] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The WEKA data mining software: an update. 11(1):10–18, 2009. 5

[23] 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. 1, 7, 8

[24] T. Kohlberger, V. Singh, C. Alvino, C. Bahlmann, and L. Grady. Evaluating segmentation error without ground truth. *Medical Image Computing and Computer Assisted Intervention (MICCAI)*, pages 528–536, 2012. 2

[25] 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. 5

[26] S. Lankton and A. Tannenbaum. Localizing region-based active contours. *IEEE Transactions on Image Processing*, 17(11):2029–2039, 2008. 1, 2, 6

[27] V. Lempitsky, P. Kohli, C. Rother, and T. Sharp. Image segmentation with a bounding box prior. In *IEEE International Conference on Computer Vision (ICCV)*, pages 277–284, 2009. 1, 3

[28] C. Li, C. Y. Kao, J. C. Gore, and Z. Ding. Minimization of region-scalable fitting energy for image segmentation. *IEEE Transactions on Image Processing*, 17(10):1940–1949, 2008. 1

[29] H. Li, F. Meng, B. Luo, and S. Zhu. Repairing bad co-segmentation using its quality evaluation and segment propagation. *IEEE Transactions on Image Processing*, 23(8):3545–3559, 2014. 2, 3

[30] D. Liu, Y. Xiong, K. Pulli, and L. Shapiro. Estimating image segmentation difficulty. In *Machine Learning and Data Mining in Pattern Recognition*, pages 484–495, 2011. 2

[31] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H. Shum. Learning to detect a salient object. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(2):353–367, 2011. 3

[32] M. Maitra, R. K. Gupta, and M. Mukherjee. Detection and counting of red blood cells in blood cell images using hough transform. *International Journal of Computer Applications*, 53(16), 2012. 3

[33] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *International Conference on Computer Vision (ICCV)*, volume 2, pages 416–423, 2001. 3

[34] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, 1979. 1, 3, 4

[35] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 3(309–314), 2004. 1, 3, 4, 6

[36] B. Settles. Active learning literature survey. Technical report, University of Wisconsin, Madison, 2010. 2

[37] S. Vijayanarasimhan and K. Grauman. Cost-sensitive active visual category learning. In *International Journal of Computer Vision*, volume 91, pages 24–44, 2011. 2

[38] C. Wah, S. Maji, and S. Belongie. Learning localized perceptual similarity metrics for interactive categorization. In *IEEE Winter Conference on Applications in Computer Vision (WACV)*, pages 502–509, 2015. 2

[39] J. Wu, Y. Zhao, J. Zhu, S. Luo, and Z. Tu. Milcut: A sweeping line multiple instance learning paradigm for interactive image segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 256–263, 2014. 1, 3