Crowd Segmentation Progress

January 27, 2017

1 Problem Formulation

The goal of quality evaluation is two-folds. Given N worker responses, find:

1. the quality of the bounding boxes (BB) drawn by worker
2. the best proposed region for a given object in an image

2 Key Assumptions and Intuitions

1. In literature, there are scoring functions that require "ground truth" and ones that are "unsupervised".
2. If a worker's response differs greatly from the ground truth, then work quality (Q_w) is low.
3. If most of the workers' response differ greatly from ground truth for a particular image, then task difficulty (D_t) should be high. As a corollary, if the spread of the worker distribution (J_i) is large, then D_t should also be high.

To compute the latent quantities Q_w,D_t, we propose an iterative EM-like algorithm, where at every step, we assume that the ground truth bounding box (BB_G) is the current estimate of the maximum likelihood region. The maximum likelihood region is constructed by adding in sub-regions from a tile-graph.

3 Model

3.1 Definitions

The model is inspired by the image-labelling model from [13] with modifications for the segmentation task as follows:

- A task is defined by an object-image pair i :
  - z_i is hidden variable that completely describes the ground truth BB from the image (e.g. set of all points in BB_G). z_j is the BB drawn by worker j.

- ϕ_j is some descriptive image-related quantity extracted from worker BB z_j. This can either be a 1-D scalar aggregate or a multidimensional quantity. (e.g. boundary complexity of the image boundary) Sometimes, these summarization functions require a comparison against the ground truth z_j, so more generically we denote ϕ_j as ϕ_{ij}
- J_i is the set of all workers j that annotated the object-image.
- D_i is the task difficulty of object-image i. The ϕ_i image summary of object i's ground truth BB z_i. The task difficulty is a dependent variable measuring how far off are most of the worker's responses (z_j) compared to z_i.

$$D_i = \sum_{j \in J_i} dist(\phi_{ij}, \phi_i) \tag{1}$$

- We assume, by definition, that task difficulty is completely a result of the image itself and independent of any worker qualities. q_{ij} is the quality of worker j for object i evaluated against z_i[1].

$$q_{ij} = dist(\phi_{ij}, \phi_i) \tag{2}$$

- Both the characteristics of the ground truth BB (ϕ_i) and the worker's ability to segment the image (q_j) would determine how the worker would perceive the image(ϕ_{ij}) and the BB drawn by the worker (z_j) would look like.

3.2 Generative process

- To determine the quality of a worker (q_j), we can take some type of aggregate over all tasks performed by the workers. Averaging is a good aggregate functions because the number of tasks performed by each worker can vary a lot:

$$q_j = \frac{1}{\text{Total \# tasks by worker j}} \sum_i q_{ij} \tag{3}$$

  This direct technique to finding q_j is used in the analysis in Sec.5.5

- Alternatively, we can also model q_{ij} as a random variable drawn from a distribution. Since the task difficulty and worker quality does not completely determine the quality of bounding box that a worker would draw next. Therefore, the distribution of ϕ_{ij} given a new bounding box z_j can be described as:

$$p(\phi_{ij}|z_j) = f(\phi_i; q_{j'}, D_i) \tag{4}$$

  where j' includes all object labelled by worker j up to object i.

[1] We can also try to model user expertise separate from q_{ij} which define only the vision-related quantities related to worker j's judgement on ϕ_{ij}, but this is less important in salient, common-object segmentation.

4 Metrics for Φ Functions

Φ is a function that summarizes a given bounding box, where $\phi_i = \Phi(z_i)$ or $\phi_{ij} = \Phi(z_i, z_j)$. When the Φ metric requires a comparison against ground truth, there are two sets of annotations that we have used for computing these metrics: gold-standard annotations cross-matched with MSCOCO (`[COCO]`) and detailed annotation boundaries drawn by me with the same web interface (`[Self]`). Note that some of the COCO annotations lack exact cross matches. The metrics computed for objects that are not in the COCO database are flagged and not used for computing the evaluation metrics. However, inexact annotations of the same object (e.g. book-labeled object with only book cover annotation) is still incorporated in the computed metrics. The evaluation metrics can be grouped into three categories:

Area-based: These methods include precision, recall, area ratio or boundary complexity. These measures take into account the intersection, $\mathcal{I} = area(z_i \cup z_j)$, or union, $\mathcal{U} = area(z_i \cap z_j)$, between the user and the ground truth bounding boxes.

$$\texttt{Precision} = \frac{\mathcal{I}}{area(z_i)} \tag{5}$$

$$\texttt{Recall} = \frac{\mathcal{I}}{area(z_j)} \tag{6}$$

$$\texttt{Jaccard} = \frac{\mathcal{U}}{\mathcal{I}} \tag{7}$$

Area ratio is a simple baseline proposed by [11] based on the intuition that larger objects should be easier to annotate than smaller objects, so larger annotations should be better than smaller ones.

$$\texttt{Area ratio} = \frac{area(z_i)}{\text{Total image area}} \tag{8}$$

Boundary-based: While precision, recall, and majority-vote are simple metrics, since they are bounded by [0,1], the metrics computed against ground truth should always be 1. In addition these projection functions do not capture the full resolution of the bounding box. [11] surveys the quality evaluation metrics for image segmentation and proposes a bipartite-matching measure based on the Euclidean distance between two BBs. First, they randomly sample m=300 points along the annotation boundary, then compute all pairwise Euclidean distance. Then, the Kuhn-Munkres algorithm is used to match together the orientation of the two annotations, and returns the assignments that yields the minimum Euclidean. Finally, the normalized score (`NME`) of an annotation i is computed as:

$$score = 1 - \frac{dist_i}{max(dist)} \tag{9}$$

where max(dist) is the maximum Euclidean distance of all the annotations computed in our dataset. Our implementation differs slightly in that we conduct a B-spline parametric interpolation of use m=50 points along the boundary rather than random sampling, in order to speed up the computation in the Munkres algorithm. These implementation details should have little effect on the Euclidean scores computed.

Another baseline used by [11] is a simple, unnormalized count of the number of points drawn by the users to construct the bounding box (`Num Points`), based on the intuition that a more carefully-annotated would result in a better annotation. Since some objects may have inherently simple geometries that could be well-annotated with a small number of control point, to account for the object's boundary complexity, one possible derived measure could be to normalize by the max number of control point) of the particular object.[2]

Contrast-based: These methods examine how close is BB to regions of contrasts detected by CV algorithms (saliency maps, Bayesian Matting) or edge detectors. A major problem when implementing these methods is group and match CV regions to BB annotations, since CV methods often yield over-segmented regions.

5 Preliminary Experiment

We ran a preliminary experiment where each HIT consisted of one annotation task for a specific pre-labelled object in the image, as shown in Fig.5. There is a total of 46 objects in 9 images from the MSCOCO dataset[6]. These objects and images are intentionally chosen so that they represent a variety of image difficulty (based on object clutter-ness) and potential logical error and level of ambiguity. The average number of objects annotations that each worker completed was 10.16. The average time to complete each HIT is 83.96 seconds and workers are compensated for 5 cents per HIT. For each object, we collected annotations from a total of 40 independent workers.

5.1 Data Observations

- **Basic statistical summary:** Since the mean is close to one, most workers make decent annotation that closely follows the ground-truth BB. While the standard deviation is large for most metrics in the unfiltered results (shown on left of Table 1), applying work quality filter significantly decreases the standard deviation, indicating that annotations with metric scores below a threshold are likely mistakes due to task ambiguity or ground-truth mismatches, rather than imprecise annotations.

- Both the number of tasks each worker takes on and average time in a task follows a Pareto-like, long-tail distribution, which is typical for crowdsourcing applications.

- **Data Fitting Procedure:** We are interested in figuring out what functional form of Φ looks like. We fitted the histograms against 84 different probability distribution functions[3], using the maximum-likelihood estimators of these distributions. Then, a Kolmogorov-Smirnov test assessed the statistical significance of whether the fitted function and the data follow the same distribution. We quantify the best fits using

[2]Since this is a constant for each object i, it would not affect the form of the J_i distribution.

[3]Most of the functions in scipy.stats:[alpha, anglit, arcsine, beta, betaprime, bradford, burr, cauchy, chi, chi2, cosine, dgamma, dweibull, expon, exponpow, exponweib, f, fatiguelife, fisk, foldcauchy, foldnorm, frechet_l, frechet_r, gamma, gausshyper, genexpon, genextreme, gengamma, genhalflogistic, genlogistic, genpareto, gilbrat, gompertz, gumbel_l, gumbel_r, halfcauchy, halflogistic, halfnorm, hypsecant, invgamma, invgauss, invweibull, johnsonsb, johnsonsu, ksone, kstwobign, laplace, levy, levy_l, loggamma, logistic, loglaplace, lognorm, lomax, maxwell, mielke, nakagami, ncf, nct, ncx2, norm, pareto, pearson3, powerlaw, powerlognorm, powernorm, rayleigh, rdist, recipinvgauss, reciprocal, rice, semicircular, t, triang, truncexpon, truncnorm, tukeylambda, uniform, vonmises, vonmises_line, wald, weibull_max, weibull_min, wrapcauchy]

All	Mean	SD	Filter>0.6	Mean	SD
Precision [COCO]	0.87	0.22	Precision [COCO]	0.93	0.069
Recall [COCO]	0.9	0.12	Recall [COCO]	0.92	0.072
Jaccard [COCO]	0.79	0.22	Jaccard [COCO]	0.86	0.084
NME [COCO]	0.94	0.12	NME [COCO]	0.96	0.055
Num Points	26	19	Num Points	26	19
Precision [Self]	0.86	0.21	Precision [Self]	0.92	0.076
Recall [Self]	0.9	0.14	Recall [Self]	0.93	0.074
Jaccard [Self]	0.78	0.22	Jaccard [Self]	0.86	0.086
NME [Self]	0.94	0.13	NME [Self]	0.96	0.053
Area Ratio	0.063	0.089	Area Ratio	0.063	0.089

Table 1: Left: Statistics for all workers; Right: for good workers only [metric≥0.6]

minimal residual sum-of-square (RSS) and the p-value resulting from the KS-test. To preserve the tails of these distributions, no filtering for selecting good workers only was done in the fitting procedure.

5.2 Overall Distribution

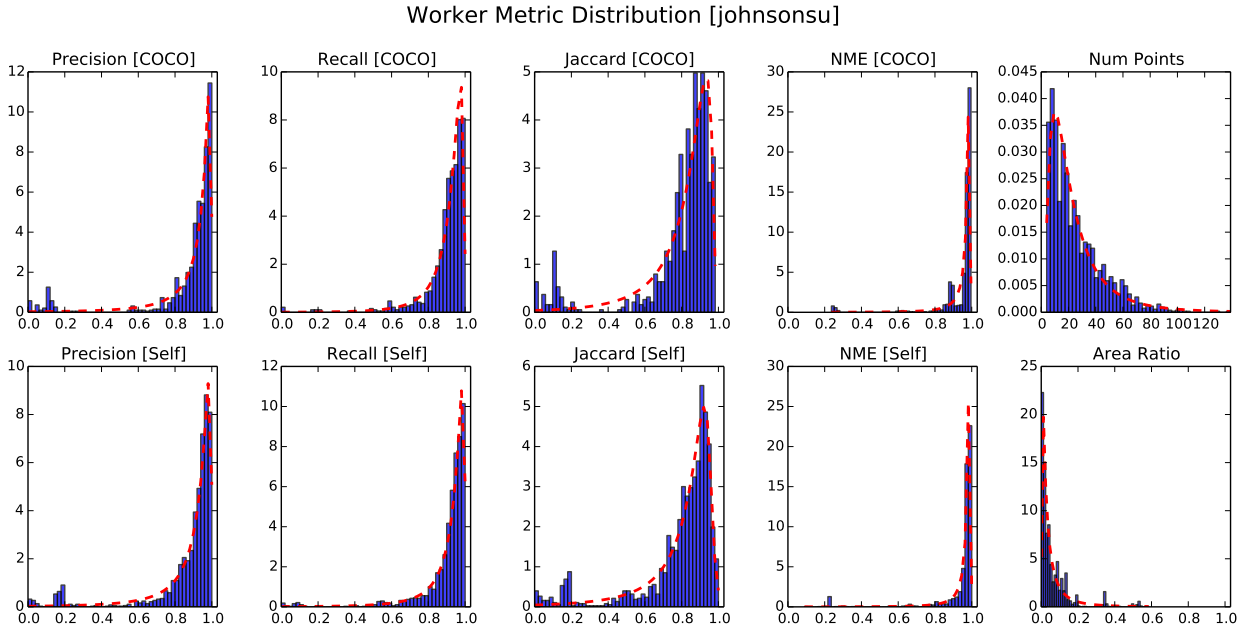

Figure 1: Normalized histogram of metric values, fitted with a Johnson SU distribution.

Overall distribution contains each of the metrics computed from all tasks submitted by all workers (N=1947). The histogram distribution (fixed bin size =50) of these metrics resembles a long-tail, exponential-decaying distribution. As shown in the best-fitting functions in Table 3, there are many pdfs that fits one metric but not another.

One particular distribution that yields the best fits for many metrics is the Johnson unbounded (SU) distribution, which we summarized in Table 4. The Johnson SU distribution is a transformed Gaussian where the data $x \mapsto \gamma + \sigma \sinh^{-1}(\frac{x-\xi}{\lambda})$, which effectively maps the typical two-parameter Gaussian to a more flexible, four parameter pdf to better account for the skewness (heavily right-skewed) and kurtosis (long-tail) of the distribution.

5.3 Object-level Distributions

Recall that J_i is the set of all workers j that annotated the object-image i, we are interested in finding out how these workers are distributed in order to deduce worker quality.

In the data fitting procedure, the bin size is an important hyperparameter. When the bin size is small, the histogram is very smoothed, so many different functional forms can be fitted. Since our data is N=40, we pick a bin size of 30. Due to the large number of J_i distributions, we conducted the fitting procedure on a smaller candidate set of more interpretable functional forms[4].

Table 5 summarizes the best functional fit for each metric, based on the average RSS across all J_i distributions. The magnitude of RSS for the fitted function of each metric is very different. If we examine a rankings, the RSS difference between the top few best-fit functions is minimal and usually contain the Johnson SU distribution, so the Johnson SU distribution is a sufficiently good description of these Φ metrics.

5.4 Task difficulty Distributions

We compute the task difficulty according to Eq.1 for all metrics and all objects. In order to check if the task difficulty agrees with our intuition on which tasks are difficult, we manually labelled our tasks into the three types of errors that workers are prone to make (task ambiguity, small area, and complex boundary)[5] We defined the easy tasks as the ones that are neither in any of the error-prone categories, and the overall category contains all 47 tasks. As shown in Fig.3 and Table 2, the easy tasks have the lowest average difficulties. Then, of all the error categories, ambiguous tasks makes the task most difficult, followed by hard-to-annotated tasks due to small area and complex boundaries.

5.5 Worker Quality Distributions

We compute the task difficulty according to Eq.2. We check that users who have high average quality scores often draw erroneous bounding boxes and that workers with low quality scores have consistently accurate bounding boxes[6]. Fig.8 shows that the functional form of the worker quality distribution most resemble a t-distribution.

[4]Based on our overall function fitting results: Gaussian, Johnson SU/SB, Cauchy, Beta, Loggamma, generalized gamma, Gompertz and t-distributions

[5]Each task can be in more than one category.

[6]Note this is opposite from the colloquial notion of "high-quality" and "low-quality", we can avoid this confusion by transforming the quality score with 1-x or 1/x.

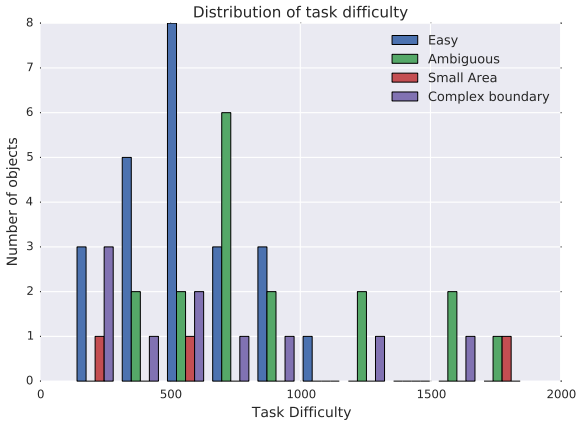

Figure 2: The distribution of task difficulty, with bin size of 10.

	Avrg task difficulty
Easy	565.96
Overall	655.52
Complex boundary	673.96
Small area	852.85
Ambiguous	903.96

Table 2: Average difficulty per category.

Figure 3: While most of the good quality workers have well-annotated BBs (such as the top left plot), some chosen examples highlights a problematic aspect of using a single metric for measuring work quality from ϕ metrics being incomplete 1-D summaries of the worker's bounding box. For instance, two BB with the same `NumPts` may resemble very different polygons (top right), and that total recall does not mean that the two bounding box perfectly aligns(top middle). Among the bad workers(bottom middle and right), we have also seen examples where a worker would consistently make the same type of mistake, possibly ignoring the instruction to annotate around the pointed object.

6 Related Works

Despite several large-scale efforts to collect image segmentation from crowds[6, 7, 10, 3], most have relied on simple area-based metrics to quantify their segmentation data quality. Research on quality evaluation of crowd segmentation often make use of heuristic approaches such as quantifying the user types and their clickstream behaviour to determine work quality[1, 9]. Others have also made use of computer vision features to determine where the annotations should be located, in order to compare against the crowd annotations [11, 8].

Dawid and Skene (1979) is the first work that used the EM algorithm to model an individual's biases and skills in the absence of ground truth data, by using a confusion matrix. Welinder and Perona (2010) proposes a general model that separately models annotator quality and the biases applied to binary, multivalued and continuous annotations. Welinder et al. (2010) develops a multidimensional concept of worker qualities and task difficulties by considering object-presence labelling as a noise generation process. The objective truth label is captured by a multidimensional quantity of task-specific measurements and deformed by worker and image related noise, the noisy vector obtained after this process is projected onto the vector of user expertise (which summarizes how well the user percieves each of these measurements), and finally the score is binarized into an inferred label. Many have extended this line of work beyond binary classifications by developing EM-like approaches that works on multiple-choice [4] as well as free-form responses [5].

However, even though EM algorithms assign probabilities regarding *how good a worker's bounding box is*, for the task of object segmentation, we are ultimately more interested the end goal of *what is the best bounding box that we can get from these data*. Even though the annotation probabilities are sufficient for determining the best binary-labels, image information such as overlapping areas would be useful and not account for in these algorithms. We suspect that this is why many area-based metrics are still more commonly used in practice than EM approaches.

References

[1] Ferran Cabezas, Axel Carlier, Vincent Charvillat, Amaia Salvador, and Xavier Giro-I-Nieto. Quality control in crowdsourced object segmentation. *Proceedings - International Conference on Image Processing, ICIP*, 2015-Decem:4243–4247, 2015. ISSN 15224880. doi: 10.1109/ICIP.2015.7351606.

[2] A. P. Dawid and A. M. Skene. Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm. 28(1):20–28, 1979.

[3] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[4] David R Karger, Sewoong Oh, and Devavrat Shah. Efficient crowdsourcing for multi-class labeling. *ACM SIGMETRICS Performance Evaluation Review*, 41(1):81, 2013. ISSN 01635999. doi: 10.1145/2494232.2465761. URL http://dl.acm.org/citation.cfm?id=2494232.2465761.

[5] Christopher H Lin, Mausam, and Daniel S Weld. Crowdsourcing control : Moving beyond multiple choice. *Uai*, pages 491—-500, 2012.

[6] Tsung Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8693 LNCS(PART 5):740–755, 2014. ISSN 16113349. doi: 10.1007/978-3-319-10602-1_48.

[7] 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 *Proc. 8th Int'l Conf. Computer Vision*, volume 2, pages 416–423, July 2001.

[8] Olga Russakovsky, Li-Jia Li, and Li Fei-Fei. Best of Both Worlds: Human-Machine Collaboration for Object Annotation. pages 2121–2131, 2015.

[9] Mehrnoosh Sameki, Danna Gurari, and Margrit Betke. Characterizing Image Segmentation Behavior of the Crowd. pages 1–4, 2015.

[10] Antonio Torralba, Bryan C. Russell, and Jenny Yuen. LabelMe: Online image annotation and applications. *Proceedings of the IEEE*, 98(8):1467–1484, 2010. ISSN 00189219. doi: 10.1109/JPROC.2010.2050290.

[11] Sirion Vittayakorn and James Hays. Quality Assessment for Crowdsourced Object Annotations. *Procedings of the British Machine Vision Conference 2011*, pages 109.1–109.11, 2011. doi: 10.5244/C.25.109. URL http://www.bmva.org/bmvc/2011/proceedings/paper109/index.html.

[12] Peter Welinder and Pietro Perona. Online crowdsourcing: Rating annotators and obtaining cost-effective labels. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops, CVPRW 2010*, pages 25–32, 2010. ISSN 2160-7508. doi: 10.1109/CVPRW.2010.5543189.

[13] Peter Welinder, Steve Branson, Serge Belongie, and Pietro Perona. The Multidimensional Wisdom of Crowds. *NIPS (Conference on Neural Information Processing Systems)*, 6:1–9, 2010. doi: 10.1.1.231.1538. URL http://www.vision.caltech.edu/visipedia/papers/WelinderEtalNIPS10.pdf.

Appendix A Best-fit summaries

metric	Function Name	RSS	D-value	p-value
Precision [COCO]	beta	9.11	0.48	1.02e-05
Recall [COCO]	loggamma	4.56	0.46	2.76e-05
Jaccard [COCO]	gompertz	13.3	0.46	2.76e-05
NME [COCO]	cauchy	51.7	0.84	1.25e-16
Num Points	johnsonsb	0.000239	1	2.16e-23
Precision [Self]	johnsonsu	6.02	0.34	0.00443
Recall [Self]	johnsonsu	5.07	0.42	0.000178
Jaccard [Self]	johnsonsu	4.57	0.28	0.0317
NME [Self]	johnsonsb	30.3	0.74	5.31e-13
Area Ratio	gengamma	27.1	0.34	0.00443

Table 3: Shapewise, the RSS is a better measure of functional fit than p-value, so we use this for evaluating the best-fitting pdf for each measure. This table summarizes the best-fitting function for each metric.

metric	RSS	D-value	p-value	ξ	λ	Shift	Scale
Precision [COCO]	10	0.36	0.0021	5.2	0.75	1	0.00011
Recall [COCO]	7	0.44	7.2e-05	5.9	1.1	1	0.00062
Jaccard [COCO]	14	0.3	0.017	5.6	1.1	0.99	0.0017
NME [COCO]	2.2e+02	0.7	1.1e-11	1.3	0.61	0.99	0.0032
Num Points	0.00044	1	2.2e-23	-6.2	1.2	0.8	0.21
Precision [Self]	6.8	0.34	0.0044	5.6	0.84	1	0.00018
Recall [Self]	5.5	0.42	0.00018	5.5	0.91	1	0.00029
Jaccard [Self]	4.6	0.28	0.032	1.6	0.95	0.96	0.039
NME [Self]	1.1e+02	0.64	7.8e-10	1.2	0.61	0.99	0.0037
Area Ratio	30	0.32	0.0089	-4.9	0.78	-0.0002	0.00012

Table 4: Johnson SU fitting coefficients.

metric	Function	RSS
Area Ratio	johnsonsu	273751.49
Jaccard [COCO]	johnsonsu	675.65
Jaccard [Self]	cauchy	250.42
NME [COCO]	johnsonsu	27439.87
NME [Self]	johnsonsu	10812.98
Num Points	cauchy	0.06
Precision [COCO]	johnsonsu	4570.24
Precision [Self]	johnsonsu	1414.98
Recall [COCO]	johnsonsu	452.79
Recall [Self]	beta	902.61

Table 5: Best functional fit for each metric, as determined by average RSS across all objects in the J_i distribution.

	P [C]	R [C]	J [C]	NME [C]	NumPt	P [C]	R [S]	J [S]	NME [S]	Area
R [Norm]	0.05	-0.27	-0.11	0.32	0.84	0.18	-0.36	-0.05	0.27	0.60
p[Norm]	0.85	0.30	0.67	0.22	0.00	0.49	0.15	0.87	0.29	0.01
R [JSU]	0.31	0.02	0.03	-0.12	-0.26	0.47	-0.19	0.61	0.51	0.27
p [JSU]	0.22	0.95	0.90	0.64	0.31	0.05	0.47	0.01	0.04	0.29

Table 6: Pearson's linear correlation coefficient when comparing the average number of points in BB drawn by all worker(as an indicator for task difficulty) and the standard deviation of the worker distribution (against `JSU` and `Norm` distributions). [C],[S] short for [COCO] and [Self].

Appendix B Data Examples

Figure 4: An example interface for the segmentation webapp can be seen here.

Visualizations for all the object annotations could be found here.

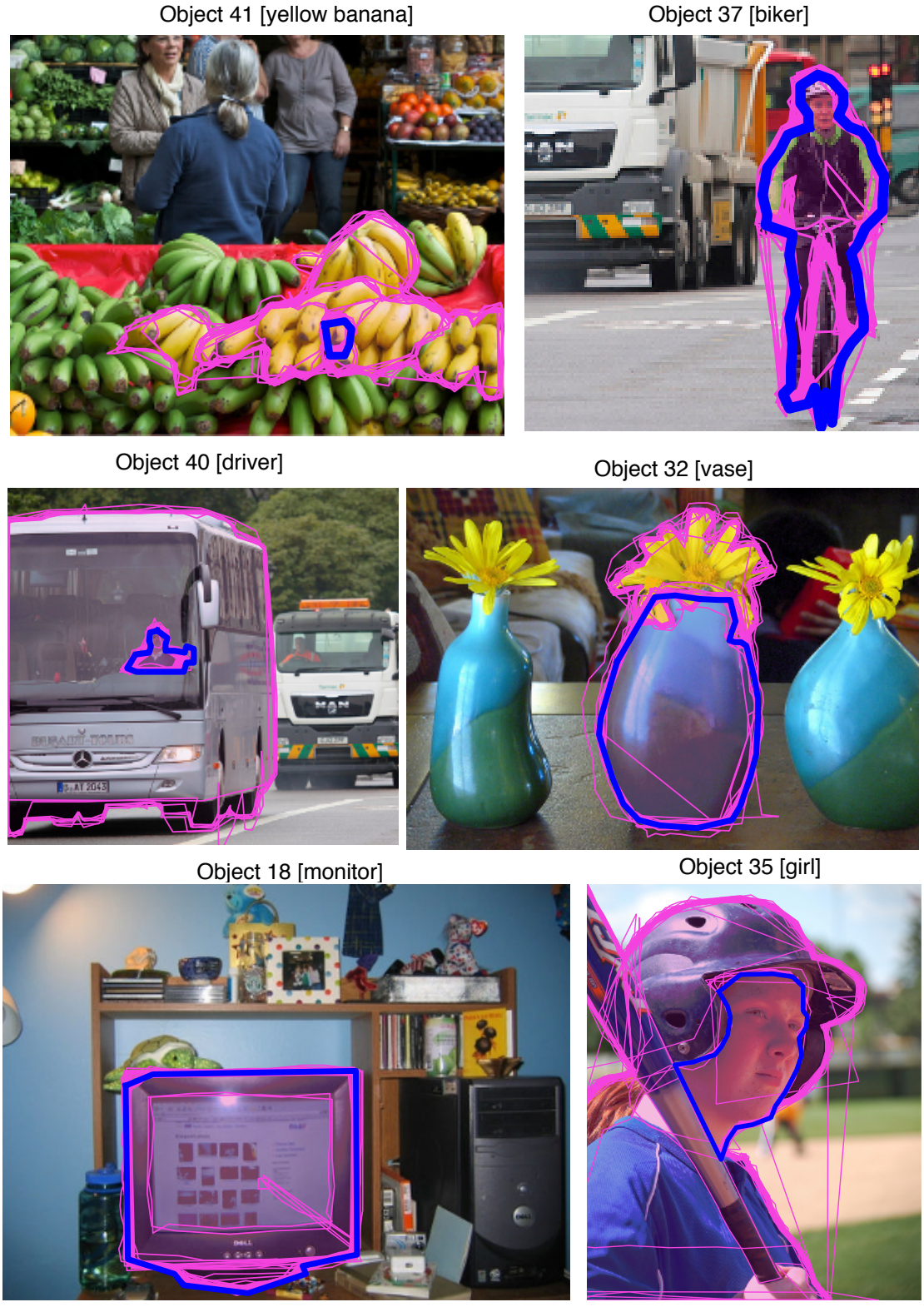

Figure 5: Selected task ambiguous object that is excluded in the task difficulty analysis.

Appendix C Additional Plots

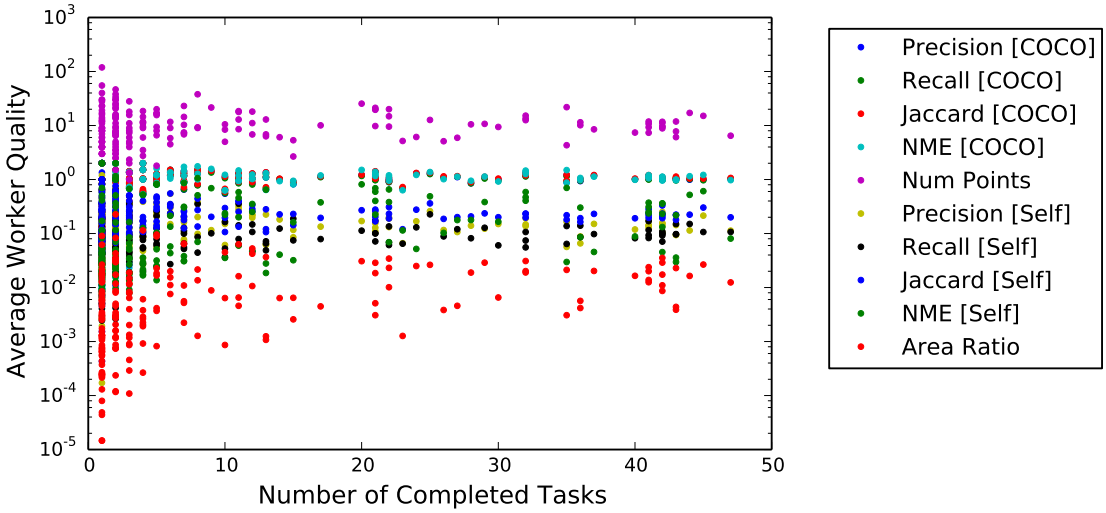

Figure 6: When a worker completes many tasks, their work quality is fairly good on average(Right region of plot). If the worker only completes a few tasks, they can either draw one very good bounding box (Bottom left), or they may draw a really bad one(Top left). In the small number of tasks case, the spread in work quality is large.

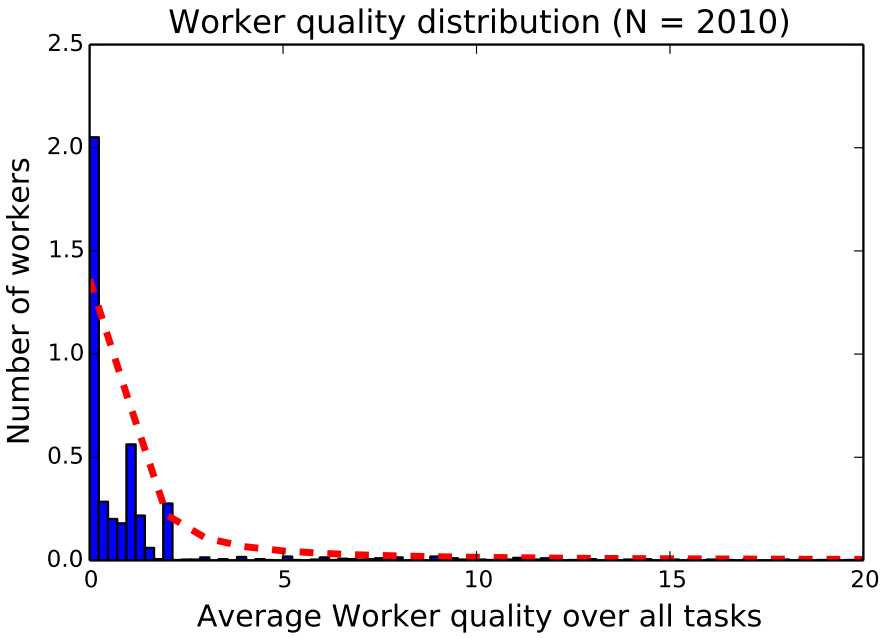

Figure 7: The normalized histogram with bin size of 500 fits best to a t-distribution, with RSS of 0.38. These trends are simmilar to the results in Figure 6c of [12].