Quality Assessment for Crowdsourced Object Annotations

Sirion Vittayakorn
svittayakorn@cs.brown.edu

James Hays
hays@cs.brown.edu

Computer Science Department
Brown university
Providence, RI, USA

Abstract

As computer vision datasets grow larger the community is increasingly relying on crowdsourced annotations to train and test our algorithms. Due to the heterogeneous and unpredictable capability of online annotators, various strategies have been proposed to "clean" crowdsourced annotations. However, these strategies typically involve getting *more* annotations, perhaps different types of annotations (e.g. a grading task), rather than computationally assessing the annotation or image content. In this paper we propose and evaluate several strategies for automatically estimating the quality of a spatial object annotation. We show that one can significantly outperform simple baselines, such as that used by LabelMe, by combining multiple image-based annotation assessment strategies.

1 Introduction

Numerous core computer vision [18] and computer graphics [6, 14] research topics rely on data sets of spatially annotated objects. Perhaps the most widely known data set is that of the annually updated Pascal VOC object detection challenge [11]. The Pascal data set was traditionally built by a dozens of skilled annotators operating with detailed instructions and expert supervision. The resulting data sets contain ∼20,000 objects in ∼10,000 images. The object annotations span 20 categories and are limited to rectangular bounding boxes. The various Pascal VOC challenges continue to advance the state of the art in object recognition, but the data sets are relatively expensive to collect, relatively limited in number of categories and amount of training data, and rectangular bounding boxes carry limited shape information.

In recent years there has been interest in larger and more diverse object data sets. Efforts such as LabelMe [4] and ImageNet [9] contain spatial annotations of thousands of object categories in hundreds of thousands of images. To build these data sets with orders of magnitude more data than expert-constructed data sets researchers must *crowdsource* the annotation effort to non-expert, minimally trained workers of heterogeneous ability. Whether such workers are paid [9] or not [4], individual annotations can not be trusted as they can for a dataset such as Pascal or Lotus Hill [2]. For this reason, numerous strategies are employed to "clean" or filter the raw annotations. Sorokin and Forsyth [1] suggest strategies such as "Gold standard", "Grading", and "Consensus". These strategies are effective, but they necessarily cost additional money (especially consensus) and add complexity to database

creation. LabelMe, on the other hand, does not explicitly use any of these strategies (although administrators will manually remove flagrant annotations). Instead LabelMe uses a simple but surprisingly effective heuristic to rank user annotations – the number of control points in each annotation.

This heuristic is an instance of what we will refer to as *annotation scoring functions*. One can think of such functions as returning a real-valued score given a user annotation and an image. Such functions are useful because they do not throw out any user annotations and allow a database to be filtered in accordance with the demands of a specific algorithm. For instance, applications such as Photo Clip Art [14] or silhouette-based detectors [12] require very accurate annotations. Fixed aspect ratio detectors like Delal and Triggs [8] or Felzenswalb et al. [13] need the rough spatial extent of each object. Finally, to learn contextual relationships one might only need the rough location of objects or one might only be concerned with whether objects co-occur in the same scene [13].

We believe this paper is the first to explore *annotation scoring functions*. We hope these methods will help researchers utilize large-scale, crowdsourced data sets in computer vision and computer graphics. An ideal annotation scoring function would rank annotations in accordance with a "ground truth" quality ranking. We will discuss how we define ground truth later in the paper. The scoring function should be unbiased. For instance, one might consider using a car detector to assess the quality of car annotations, but this circularity would bias you toward using car instances which are *already easy to recognize* and thus defeat the purpose of using larger datasets. To help avoid such biases, we examine only category agnostic scoring functions (i.e. scoring functions which consider only the spatial annotation and not the category label).

The baseline scoring function proposed by LabelMe [4] which counts control points is surprisingly effective but it does not even consider the image itself. Clearly, we should be able to score annotations more intelligently by considering whether an annotation is likely given the local or global image content. While there is little previous work on "annotation scoring functions" per se, there is considerable research on edge detection, image segmentation, and more recently generic object detection which might relate to annotation quality. In Section 3 we operationalize such techniques for use in annotation assessment and compare their performance to simple baselines. A combined scoring function leveraging multiple cues significantly outperforms any existing baseline.

1.1 Additional Related Work

The argument for effective annotation scoring functions is partly an economic argument. If money weren't an issue, collecting more annotations and involving more expert supervision would always be a solution. To make large-scale annotation economical and fast there is ongoing research into active learning methods for visual recognition which preferentially collect the most informative annotations [20], sometimes while considering the expected human annotation effort [19]. These methods are very effective but they are task dependent. To annotate a general purpose database like LabelMe or ImageNet these methods are not suitable because there is no single classification task to guide the active learning. Also, while these methods are informative about *which* annotations to collect, they don't usually offer any unique insight into which annotations to *trust*.

The image synthesis method of Sketch2photo[6] could be thought of as including an annotation scoring function. In this case the annotations or bounding boxes being assessed are not from humans but from automatic segmentation techniques that are expected to have a

high failure rate. Sketch2photo focuses on finding "algorithm friendly" images which segment easily (for instance, containing only one salient object). Our proposed methods might also have a tendency to reward "algorithm friendly" images for which the user annotation and automatic segmentation agree. Our evaluation penalizes any such bias, though.

2 Definitions and Dataset

An annotation scoring function is a function that takes an image and user's annotation and returns a real-valued score indicating the quality of that annotation. By our definition, an annotation scoring function does not consider the supposed *category* of an annotation. We do not consider the situation where an annotation is poor quality because the annotator gave it the wrong label (e.g. a user accurately segmented a dog but then labeled it a car). In our experience, this is a very rare situation. In this paper, we investigate five annotation scoring functions and have each method rank hundreds of crowdsourced user annotations. We then compare those rankings of annotation quality to a ground truth ranking using Spearman's rank correlation coefficient [16]. To make this comparison we must first build a dataset containing pairs of crowdsourced annotations with their corresponding ground truth annotation.

2.1 User Annotations

We collect spatial object annotations from LabelMe [4]. We expect such annotations to be qualitatively similar to those obtained from other sources such as Mechanical Turk. LabelMe annotations are closed polygons entered from a web-based interface. All of the scoring functions, except for the LabelMe baseline, take a binary mask as input rather than a polygon. Thus our functions are suitable to any labeling interface that produces a bounding box or segmentation. For the experiments in this paper, we collect 200 pairs of object images and user annotations from 5 categories – person, car, chair, dog, and building. These categories are picked because they are among the most common objects in LabelMe and because they are distinct in size, shape, and annotation difficulty.

2.2 Ground Truth Annotations

For these 1,000 total images we need to establish a ground truth spatial annotation in order to know how good each user annotation is. The red contour in Figure 1 shows a sample of the ground truth annotation for five objects in our dataset. Because we are assessing LabelMe annotations, our ground truth is specified in strict accordance with the LabelMe guidelines. For instance, different datasets have different rules about how to annotate occlusions. We do not think these rules had a significant impact on user annotation quality, except for the building category where user annotations violated these rules occasionally. The actual ground truth segmentation was a time-consuming, manual process. We have released this database online.

2.3 Comparing Annotations

To establish a ground truth quality score for user annotations we need to compare user annotations to our ground truth annotations. There are numerous methods in the vision literature for comparing such shapes. Perhaps the most widely used is the PASCAL VOC overlap

Figure 1: Example of the ground truth dataset from 5 categories. The red bounding box is the ground truth bounding box of object.

score [11]. However, the overlap score is not entirely satisfying for arbitrary shapes because it gives no special attention to boundaries or small protrusions. For instance, a person annotation might still be scored quite highly even if extended arms are completely left out, because those pixels are perhaps only five percent of the area of the shape. Therefore, we also use the *Euclidean distance score*, inspired by the "Shape Context" method [3], which emphasizes the boundary agreement of two annotations.

Overlap Score. In PASCAL VOC competition, algorithms are evaluated based on the overlap area of the two annotations. For two objects B_u and B_v, the overlap score is:

$$score_p = \frac{area(B_u \cap B_v)}{area(B_u \cup B_v)} \tag{1}$$

where $B_u \cap B_v$ denotes the intersection of two bounding boxes or binary masks and $B_u \cup B_v$ their union.

Euclidean Distance Score. The Euclidean distance score measures the distance between two annotation boundaries. There are numerous shape matching methods and we chose a bipartite matching method in the spirit of [3]. We sample $m = 300$ random points along each annotation boundary then calculate the Euclidean distance between all possible pairwise correspondences, resulting in an m by m matrix of Euclidean distances. Given this matrix, the Kuhn-Munkres algorithm [15] returns the m assignments which have the minimum total Euclidean distance. In Figure 2 left, the red and green points are the random points from the user and ground truth annotations, respectively, while the blue lines show the Euclidean distance between the assigned pairwise. Given these correspondences for annotations B_v and B_u where $(X_i, Y_i) \in B_u, (x_{i'}, y_{i'}) \in B_v$ we simply sum the pairwise distances according to Equation 2. Finally, we normalize these distances into [0,1]. Because larger Euclidean distances are associated with lower annotation quality, we define our Euclidean distance score as 1 minus the normalized summed distance (Equation 3).

$$dist = \sum_i \sqrt{(X_i - x_{i'})^2 + (Y_i - y_{i'})^2} \tag{2}$$

$$score_e = 1 - \frac{dist}{max(dist)} \tag{3}$$

And $max(dist)$ is the maximum Euclidean distance of that category.

2.4 Ground Truth Quality Ranking

To create our per-category ground truth ranking, we sort the user annotations by a combined score: $score = (w_1 \times score_e) + (w_2 \times score_p)$ where $score_e$ is the Euclidean distance score,
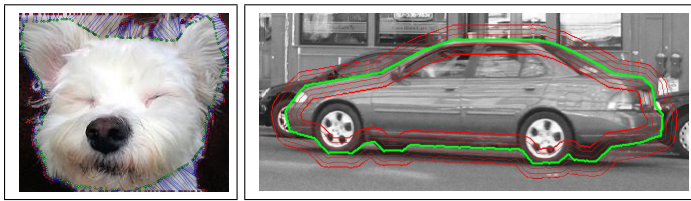
Figure 2: Left: The blue line shows the Euclidean distance between the ground truth bounding box (red points) and the user's bounding box (green points) Right: The green contour is the user's bounding box, while the red contours are the candidate object's contour base on the user's bounding box

$score_p$ is the overlap score between the user and ground truth bounding box while w_1 and w_2 were set to 0.5.

3 Annotation Scoring Functions

Given an image and annotation, we evaluate the quality of each annotation using an annotation scoring function and then rank all of the annotations within each object category. We investigate five annotation scoring functions: number of annotation control points [4], annotation size, edge detection, Bayesian matting [2] and object proposal [10].

3.1 Baselines: Control Points and Annotation Size

We start with the simple baseline used by LabelMe [4] and techniques which derive data from LabelMe such as Photo Clipart [14]. For this baseline, the quality of an annotation is proportional to the number of control points in the bounding polygon. This baseline works well – a large number of control points usually indicates that the user made an effort to closely follow the object boundary.

We also propose an even simpler baseline relating annotation size and quality of annotation. We assume that the larger an object is, the better the annotation is because it is easier for a user to annotate a large, high resolution object than a smaller one. We calculate the size baseline score as the percentage of image area occupied by the user annotation.

3.2 Edge Detection

Intuitively, a good annotation will tend to follow image edges. But this is not strictly true of most user annotations, even those which are qualitatively quite good. Users have a tendency to annotate *outside* the actual object boundary by a few pixels even when they are tracing the shape quite accurately. To accommodate for this we consider not just the user bounding polygon, but also several dilations and erosions of this bounding region as shown in Figure 2, right. For each such dilated or eroded user boundary we measure the overlap with filtered image edges. We first run Canny edge detection [5] with threshold 0.2 on the image and group the resulting edge fragments into contours. If w is the diagonal length of a user's annotation's bounding box, we discard contours which are (1) shorter than 3% of w, because such contours tend to be texture not boundaries, (2) more than 5% of w away from the user
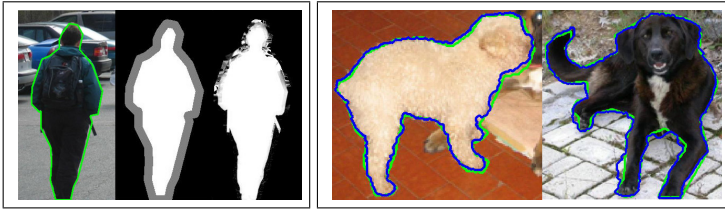
Figure 3: Left:The input image with annotation, its trimap and its alpha map from Bayesian Matting approach. The black, gray and white region in the trimap are the background, unknown and foreground region respectively. Right:The blue contours are regions which are similar to the user's bounding box (green contour) from object proposal approach

annotation, or (3) oriented more than 20 degrees away from the nearby user annotation. Finally, we find the boundary which has the most edge overlap and we assign a score to the user annotation in proportion to how much this best boundary was dilated or eroded.

3.3 Bayesian Matting

Good annotations often agree with localized image segmentation. The fourth annotation scoring function is based on the Bayesian matting algorithm [7]. Bayesian matting models the object and background color distributions with spatially-varying mixtures of Gaussians and assumes a fractional blending of the object and background colors to produce the observed pixel colors. It then uses a maximum-likelihood criterion to estimate the optimal opacity, object and background simultaneously.

The input for Bayesian matting is a *trimap* with three regions: "background", "object" and "unknown" where the background and object regions have been delineated conservatively. We construct these three regions based on the user annotation. The region outside the user annotation is constrained as background. The inside region is constrained as object. All pixels within a small and fixed distance, 3% of the length of user's bounding box's diagonal, of the user boundary are marked unknown (see Figure 3, left). Bayesian matting returns an *alpha map* with values between 0 to 1 where 0 means background and 1 means object. The fractional opacities, α, can be interpreted as a confidence of that pixel belonging to the object. We use these α values to compute an annotation score as follows: $score = \sum \alpha_{in}/area_{in} - \sum \alpha_{out}/area_{out}$ where α_{in} and α_{out} are the opacity of each pixel within the unknown region inside and outside the annotation respectively while $area_{in}$ and $area_{out}$ are the area of the unknown region inside and outside the annotation.

3.4 Object Proposal

The last annotation scoring function is based on the object proposal approach by Endres and Hoiem [10]. In their work, they propose a category-independent method to produce a ranking of potential object regions. Given an input image, their approach generates a set of segmentations by performing graph cuts based on a seed region and a learned affinity function. Then these regions are ranked using structured learning based on various cues. Top-ranked regions are likely to be good segmentations of objects. If the user annotation is similar to one of the top ranked object proposals that suggests the annotation may be

Catergory	Rank correlation					
	Points	Size	Edge	Bayesian	Proposal	Final
Car	0.5216	0.4356	0.5972	0.3848	0.0817	**0.5999**
Chair	0.6758	0.6519	0.6132	0.6780	0.0190	**0.6947**
Building	−0.3874	0.4271	0.4055	0.2030	0.0386	**0.5214**
Person	0.5503	0.4386	0.5716	0.7036	0.0394	**0.7072**
Dog	0.6070	0.2367	0.6932	0.6503	0.0468	**0.7689**
Average	0.3935	0.4380	0.5761	0.5239	0.0232	**0.6584**

Table 1: The rank correlation between ground truth ranking and other rankings

high quality. To score a user annotation, we find the object proposal with smallest Euclidean distance and overlap score to the user annotation. The score is the rank of that object proposal divided by the total number of object proposals. We also tried to have the object proposal method [10] directly score the segmentation provided by the user, and not explore multiple segmentations, but the performance was worse.

4 Comparing Ranking

At this point we have a ground truth ranking for the user annotations within each category as well as a ranking from each scoring function. We measure how well each ranking agrees with the ground truth using Spearman's rank correlation coefficient or Spearman's rho, ρ [16]. The n raw scores X_i, Y_i are converted to ranks x_i, y_i and the differences $d_i = x_i - y_i$ between the ranks of each observation on the two variables are calculated. If there are no tied ranks, then ρ is given by:

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)} \tag{4}$$

An increasing rank correlation coefficient implies increasing agreement between rankings. The coefficient is inside the interval [-1, 1] and can be interpreted as 1 if the agreement between the two rankings is perfect, 0 if the rankings are completely independent, and -1 if one ranking is the reverse of the other. We also measured the rank correlation using Kendall rank correlation coefficient [13], but there were no changes in the relative performance of the scoring functions so we do not report these numbers.

5 Results

Given the input image and its corresponding annotation, the annotation scoring function returns a score corresponding to the estimated quality of that annotation. We then generate the overall annotation ranking from each scoring function and evaluate these rankings by calculating the rank correlation against the ground truth ranking. Table 1 shows the rank correlations for each annotation scoring function broken down by category. "Building" stands out as the most difficult category for most scoring functions. We believe this is because LabelMe users did not reliably follow the annotation rules – they sometimes included multiple buildings in a single annotation, or handled occlusion incorrectly. This led to a significant disparity between some user and ground truth annotation pairs.

Bounding box / Number in the ranking (out of 200 images)					
Ground truth	118	85	21	1	186
Number of points	143	106	6	5	25
Annotation size	70	107	35	3	166
Bayesian matting	84	73	38	19	155
Edge detection	109	101	32	77	60
Object proposal	3	96	113	99	43
Final	117	92	29	18	159

Figure 4: Selected user annotations and their rank according to each scoring function.

These results show that the *number of control points* is indeed a good predictor of annotation quality except for the building category. We think that this metric is uniquely bad for the building category because many buildings are rectangular, thus many accurate annotations have only 4 control points. Interestingly, the even simpler *annotation size* baseline performs about as well as the control point baseline and is more directly applicable to annotation protocols that don't use polygons such as freehand lassoing. Both the *Bayesian matting* and *edge detection* scoring functions have a high rank correlation. We believe the Bayesian matting approach handles foreground occlusions poorly and thus does not perform well on the building category. We expected the *object proposal* scoring function to perform better. But, in fact, the algorithm is answering a somewhat different question than what we are interested in. The object proposal evaluates *how likely is this segment to be an object?* and the question we are interested in is *how accurately annotated is this object segment?*. We know that all user annotations are object segments – none of the annotations are so adversarial as to contain no object. The object proposal method may be intentionally *invariant* to annotation quality, because the automatic segmentations that it considers cannot be expected to be particularly accurate.

Finally, we investigate the performance of combinations of scoring functions. It is simple to "average" multiple rankings by converting a rank to a real valued number, averaging, and then re-ordering. We tried numerous combinations and found that none could exceed the performance of the Bayesian matting and edge scoring functions together. This "final" combination scores the highest for every category. We were surprised that the baseline scoring functions did not reinforce the more advanced, image-based functions. We also visualize specific annotation instances are ranked by each function in Figure 4 and the rankings that result from each scoring function in Table 2.

6   Test Scenario: Object Classification

We would like to confirm that using annotation scoring functions to filter crowdsourced training examples can lead to increased performance in secondary vision or graphics tasks. In particular, we examine a simple 5-way object classification task. The test set is made up of 50 cropped objects from each of the five categories of our database. We create two training sets of LabelMe annotated objects. For each category, training set I contains 50 random instances while training set II contains the 50 top ranked instances according to our combined annotation scoring function. There is no overlap between the test set and

Ranking method	Example of best ranked	Example of worst ranked
Ground truth		
Number of points		
Annotation size		
Edge detection		
Bayesian Matting		
Object proposal		
Final		

Table 2: The five highest and lowest ranked user annotations according to the ground truth ranking and each annotation scoring function.The green and red bounding boxes are the user's and ground truth bounding boxes respectively, while the blue contours are the object's contour from edge detection approach.

Category	Classification Accuracy	
	Training set I	Training set II
Car	79.9%	**88.6%**
Chair	75.5%	**82.5%**
Building	74.7%	**79.7%**
Person	85.3%	**94%**
Dog	74%	**82%**
Average	77.9%	**85.3%**

Table 3: The classification accuracy of each category

either training set. The test set objects are masked (background set to zero) according to the ground truth annotations. The training set objects are likewise masked according to the user annotations. We represent masked object crops with the gist descriptor [7]. This is a primitive but straightforward way to encode both object shape and appearance. We train one-vs-all linear SVMs for each category and show quantitative results in Table 3. The results show that using our scoring function as a filter increases the classification rate in every category and decreases the misclassification rate by 36% on average.

7   Conclusion

In this paper we rigorously show that numerous annotation scoring functions, some very simple, can produce annotation rankings that are reasonably similar to the ground truth annotation quality rankings. We propose new annotation scoring functions and show that, in isolation or combination, they outperform the simple LabelMe baseline. To evaluate these scoring functions we produced an extensive database with 1,000 pairs of crowdsourced annotations and meticulous ground truth annotations. As a sanity check, we show that user annotations ranked highly by our annotation scoring functions are more effective training data than random user annotations in a simple object classification task. We share our dataset and look forward to the development of better annotation scoring functions which will make crowdsourced annotations easier to leverage.

References

[1] David Forsyth Alexander Sorokin. Utility data annotation with amazon mechanical turk. *First IEEE Workshop on Internet Vision at CVPR 08*, 2008.

[2] X. Yang B. Yao and S.-C. Zhu. Introduction to a large scale general purpose ground truth dataset: methodology, annotation tool, and benchmarks. *EMMCVPR 2007*, pages 169–183, 2007.

[3] Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, April 2002.

[4] K. P. Murphy C. Russell, A. Torralba and W. T. Freeman. Labelme: a database and web-based tool for image annotation. *International Journal of Computer Vision*, 77: 157–173, May 2008.

[5] John Canny. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-8(6):679 –698, nov. 1986. ISSN 0162-8828. doi: 10.1109/TPAMI.1986.4767851.

[6] Cheng M. Tan P. Shamir A. Hu S. Chen, T. Sketch2photo: Internet image montage. *ACM Transactions on Graphics*, 28(5), December 2009.

[7] Yung-Yu Chuang, Brian Curless, David H. Salesin, and Richard Szeliski. A bayesian approach to digital matting. In *Proceedings of IEEE CVPR 2001*, volume 2, pages 264–271. IEEE Computer Society, December 2001.

[8] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 1: 886–893, 2005. ISSN 1063-6919. doi: http://doi.ieeecomputersociety.org/10.1109/CVPR.2005.177.

[9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR09*, 2009.

[10] Ian Endres and Derek Hoiem. Category independent object proposals. *Computer Vision – ECCV 2010*, 6315:575–588, 2010.

[11] M. Everingham, L. Van 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, June 2010.

[12] Abhinav Gupta, Jianbo Shi, and Larry S. Davis. A "shape aware" model for semisupervised learning. In *of Objects and Its Context, Proc. Conf. Neural Information Processing Systems*, 2008.

[13] M. G. Kendall. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93, June 1938. URL http://www.jstor.org/stable/2332226.

[14] Jean-François Lalonde, Derek Hoiem, Alexei A. Efros, Carsten Rother, John Winn, and Antonio Criminisi. Photo clip art. *ACM Transactions on Graphics (SIGGRAPH 2007)*, 26(3):3, August 2007.

[15] J. Munkres. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics*, 5:32, 1957.

[16] Jerome Myers. *Research Design and Statistical Analysis*. Lawrence Erlbaum Associates, Hillsdale, 2003. ISBN 0805840370.

[17] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *Int. J. Comput. Vision*, 42(3):145–175, 2001.

[18] D. McAllester D. Ramanan. P. Felzenszwalb, R. Girshick. Object detection with discriminatively trained part-based models. *IEEE Pattern Analysis and Machine Intelligence (PAMI)*, 2009.

[19] S. Vijayanarasimhan and K. Grauman. What's it going to cost you?: Predicting effort vs. informativeness for multi-label image annotations. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2262 –2269, june 2009. doi: 10.1109/CVPR.2009.5206705.

[20] S. Vijayanarasimhan and K. Grauman. Large-scale live active learning:training object detectors with crawled data and crowds. *Computer Vision and Pattern Recognition, 2011.*, april 2011.