What If Localization Worked?

Anonymous CVPR submission

Paper ID ****

Abstract

Real world computer vision systems typically have some intrinsic value in their underlying business use. Serving the the right image in a search result ad might be worth $0.001 and counting nuclear particles in material images might be worth $10,000. In general, we want to build systems which produce sufficiently accurate results within a given budget. Although an interaction with human workers can improve accuracy in many algorithms, it also increases the cost. Most computer vision research focuses on purely automated algorithms, arguing that human labor is much too expensive to be included as subroutines in operational algorithms. In this work, we put this argument into perspective by investigating joint algorithms using computers and human labor. In particular, we focus on how different degrees of human involvement effect the algorithm's accuracies. We focus on the representative computer vision task of localization, however, our general methodology can similarly be applied to other tasks, e.g., Object Detection or Image Matching. We introduce several general strategies for combining existing computer vision algorithms with human labor, e.g. (To be filled). We evaluate our results on three reference data sets, i.e., UIUC Cars, Caltech Pedestrian and Street View House numbers, and show how the accuracy of the algorithms scale with varying degrees of human involvement. Finally, we provide an outlook on what computer vision applications become possible, if localization works.

1. Introduction

2. Localization

A place to describe

1. definition of what we understand it is
2. Where localization fits into a typical classification pipeline, which helps us motivate why we chose to focus on this problem in particular
3. state of the art
4. data sets
5. applications

Object localization is one of the most obvious parts of any computer vision application. It involves two basic tasks namely semantic segmentation and object detection. Semantic segmentation is the task of labelling the pixels of an image depending on their semantic category while object detection produces a rough location (bounding boxes) of all the instances that belong to a set of objects of interest. Our research focuses more on object detection. Object detection allows us to count the number of objects precisely, which is not always possible from semantic segmentation. Where does localization fit into a typical classification pipeline? This helps us motivate why we chose to focus on the problem of localization in particular(to be done) These are some previously proposed papers on object localization using different techniques [**?**, **?**]. Although these papers provide various impressive techniques, we through our paper would like to improve upon the accuracy of the localization algorithm by incorporating HPU component with it.

TODO: We're going to have to be extremely consistent with how we use our terminology. "Detection"? "Localization"? What's the difference? After we decide this (and this section is probably a good place to choose), we should do a Ctrl+F on all instances of both words and change them to match.

3. Data Sets

1. in detail the picked data sets (refer to previous section why they are important.)
2. the measure of accuracy on those data sets

3.1. UIUC Cars

For our first example, we chose the classic UIUC cars dataset, introduced in [1, 5]. This dataset makes a particularly good baseline because it is well-established and

because there is no recognition task after the initial detection/localization step. This dataset allows us to measure the performance of a simple single-class open-set detection task. The 550 positive samples in the test set are side views of cars, some of which are partially occluded, and the 500 negative training samples contain natural scenes, various other vehicles, etc.

On this dataset, performance is evaluated using standard precision/recall measures. F-measure and absolute number of false positives is also typically reported. In the single-scale case, the output bounding box is 100×40px, and it counts as a correct detection if the center lies within a certain ellipse of the groundtruth's bounding box center. Since there may be many cars per image; statistics are aggregated over each individual car in the test set, not per image. The original parts-based representation in [1] achieves about 77.58% F-measure. TODO: What is the state of the art?

3.2. Caltech Pedestrian Detection Benchmark

The Caltech Pedestrian Detection Benchmark, introduced by [2], is a challenging pedestrian detection dataset. This set is several orders of magnitude larger than the UIUC Cars set, containing 350,000 groundtruth pedestrian bounding boxes. The research that uses this dataset is summarized by [3].

To evaluate performance, this dataset asks authors to calculate a standard ROC curve of their detection results. However, in the open set "self-driving car looking for pedestrians" scenario, the amount of negative data is potentially infinite, which means plotting TP/TR is not very insightful. Instead, authors report miss rate versus number of false positives per image ("*FPPI*"). This ROC curve is summarized as the "*log-average miss rate,*" computed by averaging the miss rate at nine FPPI values along logarithmically-spaced intervals [3].

A detection counts as correct if its intersection-over-union score is greater than 0.5, meaning $\frac{\text{AREA}(A \cap B)}{\text{AREA}(A \cup B)} \geq 0.5$, where A and B are the detected and groundtruth bounding boxes respectively.

3.3. Street View House numbers

[4]

Problem: The first paper that uses this dataset [4] only reports classification accuracy, which is separate from the precision/recall tasks. It's unclear to me where they're assuming they have groundtruth segmentation and where they do not start from the crop set. We might be able to spin the story as "Well what if it wasn't already solved? How would the existing algorithms perform if they didn't already have good crops? And how can we minimize this performance drop? This is a hard problem and nobody's thinking about it". Or we could just pick a different dataset. Not sure.

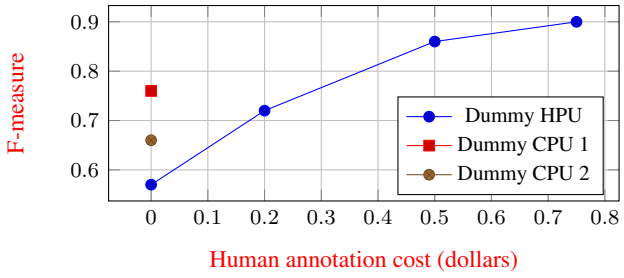

Figure 1. A graph showing how using crowdsourcing to adjust localization errors can improve accuracy. F-measure is shown on the Y-axis and dollars is shown on the X-axis. See Sec. 3.1 for details.

Question for you guys: did anyone find previous work that measures localization accuracy on this dataset?

4. Computer-Human Localization

A place to describe

1. Our different HCOMP algorithms
2. Examples
3. Evaluations (the famous plots we always talk about)

4.1. UIUC Cars

4.2. Caltech Pedestrians

4.3. Street view house numbers

5. Applications

A place to describe

1. Now that we've shown that we can improve accuracy, what other applications can we enable?
2. Graph that shows, human labour cost vs, profit of company/test... (the graph is generated by intersecting(or whatever maximization is needed between cost/accuracy and value/accuracy)

6. Conclusion

References

[1] S. Agarwal and D. Roth. Learning a sparse representation for object detection. In *Computer VisionECCV 2002*, pages 113–127. Springer, 2002. 1

[2] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: A benchmark. In *CVPR*, June 2009. 2

[3] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: An evaluation of the state of the art. *PAMI*, 34, 2012. 2

[4] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, volume 2011, page 5. Granada, Spain, 2011. 2

[5] A. A. Shivani Agarwal and D. Roth. Learning to detect objects in images via a sparse, part-based representation. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 26, pages 1475–1490. 2004. 1