Optimal Correction Cost for Object Detection Evaluation

Mayu Otani
CyberAgent, Inc.

Riku Togashi
CyberAgent, Inc.

Yuta Nakashima
Osaka University

Esa Rahtu
Tampere University

Janne Heikkilä
University of Oulu

Shin'ichi Satoh
CyberAgent, Inc.

Abstract

Mean Average Precision (mAP) is the primary evaluation measure for object detection. Although object detection has a broad range of applications, mAP evaluates detectors in terms of the performance of ranked instance retrieval. Such the assumption for the evaluation task does not suit some downstream tasks. To alleviate the gap between downstream tasks and the evaluation scenario, we propose Optimal Correction Cost (OC-cost), which assesses detection accuracy at image level. OC-cost computes the cost of correcting detections to ground truths as a measure of accuracy. The cost is obtained by solving an optimal transportation problem between the detections and the ground truths. Unlike mAP, OC-cost is designed to penalize false positive and false negative detections properly, and every image in a dataset is treated equally. Our experimental result validates that OC-cost has better agreement with human preference than a ranking-based measure, i.e.*, mAP for a single image. We also show that detectors' rankings by OC-cost are more consistent on different data splits than mAP. Our goal is not to replace mAP with OC-cost but provide an additional tool to evaluate detectors from another aspect. To help future researchers and developers choose a target measure, we provide a series of experiments to clarify how mAP and OC-cost differ.*

1. Introduction

Evaluation measure is an important factor determining the direction of the algorithm development. Most object detection benchmarks adopt the mean average precision (mAP) as their primary evaluation metric, and, therefore, great efforts are made to achieve higher mAP scores. While much research relies on mAP, we may not be completely aware of the consequences of optimising mAP.

Mean average precision is a ranking measure used in the information retrieval community [20]. Originally, the VOC challenge adopted mAP and included it into the evaluation protocol for object detection [7]. In this evaluation protocol, all detected instances are ranked in the order of their confidence scores. Then, the average precision (AP) for each object category is calculated from the precision/recall curve of the ranked instances. Mean average precision summarizes the individual APs by averaging them across all categories. Evaluation using mAPs views object detection as a task to rank detected instances for each category. However, the range of real-world applications of detection algorithms is broad, and the ranking measures may not always be the appropriate objective to be optimized.

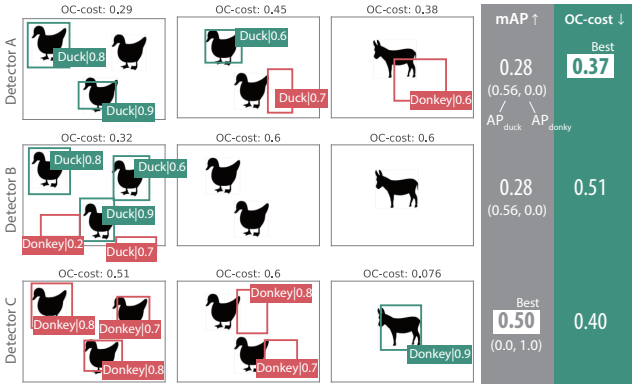

Figure 1. Toy example of three detectors and corresponding mAP and OC-cost. Higher is better in mAP, and lower is better in OC-cost. Top and middle: mAP does not treat each image equally. Detector A and B get the same mAP even though detector B does not produces any detections in the two images. Bottom: mAP does not penalizes incorrect detections ranked lower than correct ones.

Figure 1 illustrates the characteristic behaviors of the mAP measure. Suppose we have three detectors A, B, and C. Each detector attempts to find ducks and donkeys from a database consisting of three images. Detectors A and

B result in the identical mAP scores, although detector B completely ignores two of the three images. This example demonstrates that mAP does not treat images equally, and a detector does not get penalized for not producing any detections in some images in the dataset. Consistent operation on variety of images is a critical property in many real-world applications. For example, in autonomous driving, neglecting the performance in rare scenes may lead to serious risks. However, mAP does not capture such local performance drop. One remedy to mitigate this problem is to compute mAP for individual images by considering each image as a dataset, which has only one sample.

There is another issue in ranking measure-based evaluation. The example of detector C illustrates how mAP ignores some types of incorrect detections, which is non-intuitive. Detector C is an extreme example that only detects donkeys. Detector C produces many incorrect donkey detections, however, mAP does not penalize the incorrect detections as the detections are ranked lower than the correct one. Although the mAP's behavior is reasonable for evaluation of a ranking problem such as content-based image retrieval, some applications need different type of evaluation. For example, services like an on-demand visual recognition API, where various users independently upload their images, have to provide consistent performance for a wide range of images. For such services, the detection accuracy in the image level is more important than the per-class ranking performance over the entire dataset.

There are also several problems in the mAP's implementation. In mAP, each prediction is assigned to a ground truth to determine if the prediction is successful or not. However, the assignment is obtained in a greedy fashion, which the obtained solution may not be optimal. Non-optimal assignment may underrate detection results. Furthermore, mAP commonly uses thresholding on the intersection over union (IoU) scores to determine the success or failure of each detection. Prior research has shown that due to this thresholding, mAP does not reflect how well the predicted bounding box localizes the ground-truth instance [15]. Lastly, in mAP, classification is more critical than the localization quality. All detections are first grouped based on the predicted category, and misclassifications are considered as complete failures regardless of their localization quality.

In this paper, we propose a new evaluation measure called *Optimal Correction Cost* (OC-cost) that aims to evaluate the detection accuracy at the image level. Different from mAP that evaluates ranked instances detected in a whole dataset, OC-cost evaluates detection result for a single image, and the score is independent from other images. Specifically, we evaluate the detection performance using the cost of correcting detections to the ground truths. We expect that our evaluation measure better suits applications where image-level detection accuracy is critical.

To address the aforementioned problems, we formulate the computation of OC-cost as an optimal transportation problem. For every detection and ground-truth pair, we define a unit correction cost that consists of a classification and localization cost. Given the pair-wise correction costs, we find the globally optimal assignment that minimizes the total cost by solving an optimal transportation problem. This approach, inspired by the previous work [8], avoids non-optimal assignments of detections to ground truths and IoU thresholding. The approach also allow users to balance classification and localization assessment. We explore the potential of the optimal transportation cost as an alternative evaluation measure for detection tasks and re-define it as an evaluation measure.

Our contributions are summarized as follows:

- We develop an alternative evaluation measure for object detection tasks. Unlike mAP, which evaluates the performance of instance ranking, ours evaluates image-level detection accuracy. Image-level evaluation is suitable for some applications where a detector is expected to work consistently over various images.
- We conduct a series of experiments to illustrate the behavior of OC-cost and how OC-cost differs from a ranking measure-based evaluation, *i.e.*, mAP. We also demonstrate that our measure is useful for developing detectors by tuning post-processing parameters.

2. Related Works

PASCAL VOC challenge introduced mAP for object detection evaluation [7]. Given a ranked list of all detected instances, a precision/recall curve is computed. mAP summarizes the precision/recall curve by computing the mean precision at equally spaced recall values. Each instance is labeled as true positive if the instance's IoU is larger than a predefined threshold. However, the binary judgment based on thresholding lacks the capability to describe the localization quality. To alleviate the problem, a variant of mAP is proposed for COCO object detection task. COCO-style mAP [14] averages mAPs over multiple IoU thresholds ranging from 0.5 to 0.95.

Some previous works have proposed fixed measures to mitigate the limitations of mAP. LRP [15] is designed to distinguish the characteristics of precision/recall curves. LRP also introduces the average IoU of true positive instances in the measure to reflect localization quality. Another research group pointed out that, due to a certain implementation of mAP, AP is not category independent [5]. Specifically, the limitation on detections per image can eliminate detections of rare categories with low confidence scores. As a result, mAP can be substantially degraded. They proposed a fixed evaluation protocol that limits the number of detections per category instead of limiting detections per image. They also

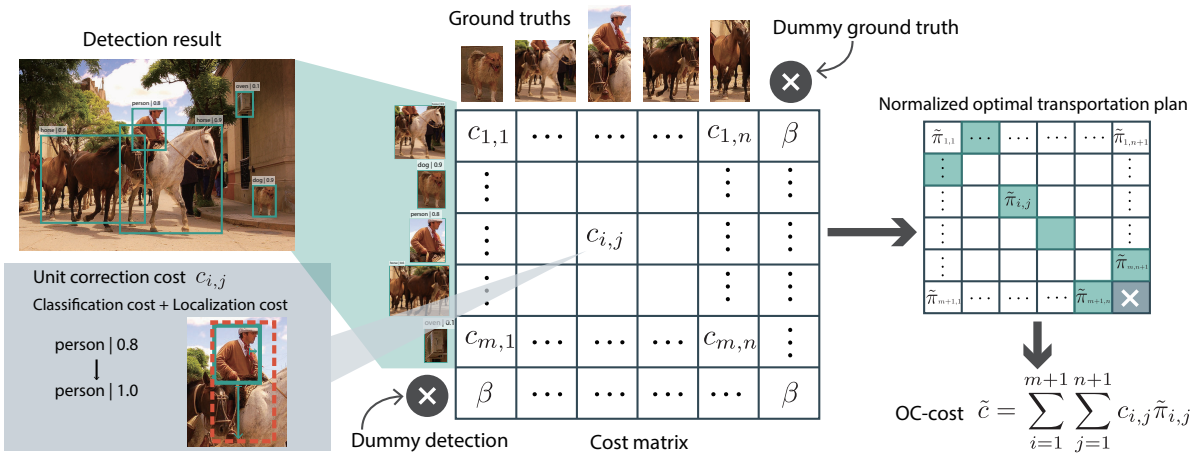

Figure 2. Overview of OC-cost. Given a set of detections and ground truths, we construct a cost matrix. Each element represents the cost of correction for a pair of a detection and a ground truth. We obtain optimal assignment of the detections to the ground truths by solving an optimal transportation problem. Based on the assignment, we aggregate the correction costs as the measure of detection performance.

proposed a variant of mAP, which evaluates cross-category ranking. Their mAP variant is employed by LVIS'21 [9]. TIDE tries to reveal how a detector fails by computing mAP drop caused by certain types of errors, *e.g.*, localization error and misclassification [11]. Probabilistic object detection indicates another direction of evaluation [10]. They proposed a new format of detection that requires to report uncertainty of detections. Their measure evaluates if the detector can accurately estimate the uncertainty of detection.

Our measure is inspired by two prior works. The first one is building detection evaluation [16]. The evaluation measure matches predicted segments of buildings and references by solving a bipartite graph matching problem, then computes a shape-aware distance between matched segments. The second work is a recently proposed learning method for object detection [8]. Their loss function is computed by solving an optimal transportation problem between predicted anchors and ground truths. Their loss function aims to promote the proper assignment of anchors to supervision during training. Since this loss function is not designed to evaluate the final detection results, it is not straightforward to use their formulation for evaluation. Their formulation assumes one-to-many matching where each ground truth can be assigned to multiple anchors, and the loss values for different samples are not comparable. To extend the idea to an image-level detection evaluation, we reformulate the optimal transportation problem.

3. Optimal Correction Cost-based Measure

For image-level evaluation, we consider costs to correct detections as a performance measure. Figure 2 illustrates the overview of our measure. We assess the cost of correction for every detection and ground truth pair. To compute the cost, we evaluate classification and localization errors. We then construct a cost matrix consisting of the pairwise costs. To find an optimal assignment of the detections to the ground truths, we solve an optimal transportation problem on the cost matrix [1], then compute the OC-cost on the assignment. To aggregate OC-costs on a dataset, we average the image-level OC-costs over all images.

3.1. Optimal Transportation Problem

We formulate the problem of finding assignment of detections to ground truths for correction cost assessment as an optimal transportation problem. The goal of an optimal transportation problem is to find an optimal transportation plan to move goods from a collection of suppliers to a collection of demanders. Suppose there are m suppliers and n demanders. The supplier i holds s_i units of goods, and the demander j needs d_j units of goods. Transporting a unit of goods from a supplier i to a demander j costs $c_{i,j}$ which constructs a pair-wise cost matrix. The goal is to transport all goods at the minimal total cost. Precisely, we consider the following optimization problem to find the optimal transportation plan π^*:

$$\pi^* = \arg\min_{\pi} \sum_{i=1}^{m} \sum_{j=1}^{n} c_{i,j} \pi_{i,j}, \tag{1}$$

$$\text{s.t.} \sum_{i=1}^{m} \pi_{i,j} = d_j, \sum_{j=1}^{n} \pi_{i,j} = s_i, \tag{2}$$

$$\sum_{i=1}^{m} s_i = \sum_{j=1}^{n} d_j, \tag{3}$$

$$\pi_{i,j} \geq 0 \ (i = 1, \ldots, m, \ j = 1, \ldots, n). \tag{4}$$

The exact solution can be obtained with linear programming. In our experiments, we use an off-the-shelf solver[1].

3.2. Unit Correction Cost

We construct a pair-wise cost matrix each of which element represents a cost to correct a detection to a certain ground truth. We call this a unit correction cost. Let b_i be the i-th bounding box and l_i be the bounding box's category label. p_i is a confidence score for the i-th detection. A unit cost to correct one detection i to a ground truth j is a weighted sum of localization and classification costs as:

$$c_{i,j} = \lambda c_{\mathrm{loc}}(b_i, b_j) + (1-\lambda) c_{\mathrm{cls}}(p_i, l_i, l_j), \tag{5}$$

where $\lambda \in [0, 1]$ is a hyper-parameter to balance localization and classification costs. If the downstream task is localization error sensitive, *e.g.*, autonomous driving, larger λ is recommended.

We define a unit localization cost $c_{\mathrm{loc}}(b_i, b_j) \in [0, 1]$ as

$$c_{\mathrm{loc}}(b_i, b_j) = \frac{1 - \mathrm{GIoU}(b_i, b_j)}{2}, \tag{6}$$

where GIoU is the generalized IoU [19]. When two bounding boxes are identical, the unit localization cost is zero. Without IoU thresholding, our proposed measure smoothly changes along with the localization quality.

A classification cost is defined as

$$c_{\mathrm{cls}}(p_i, l_i, l_j) = \begin{cases} \frac{1-p_i}{2}, & \text{if } l_i = l_j, \\ \frac{1+p_i}{2}, & \text{otherwise.} \end{cases} \tag{7}$$

For correctly labeled detections, higher confidence scores are rewarded more, while misclassification is heavily penalized. Note that the unit cost is the sum of localization and classification costs, we evaluate the localization quality of misclassified detections.

There are two types of erroneous detections. A false positive detection refers to incorrectly detecting an instance, *e.g.*, due to detection in background, misclassification or poor localization. Following Pascal VOC and COCO, when there are multiple detections of the same instance, we consider the redundant detections except the best one as false positives. A false negative refers to the error of not detecting a ground truth instance. While finding the optimal assignment, false positive and negative detections have to be properly handled because involving false positives or negatives in the assignment can break good matches of detections and ground truths. To this end, we introduce a dummy detection and a dummy ground truth. False positive detections are represented by a transportation from a detection to the dummy ground truth, and false negatives are represented by transportation from the dummy detection to a ground truth.

[1]https://PythonOT.github.io/

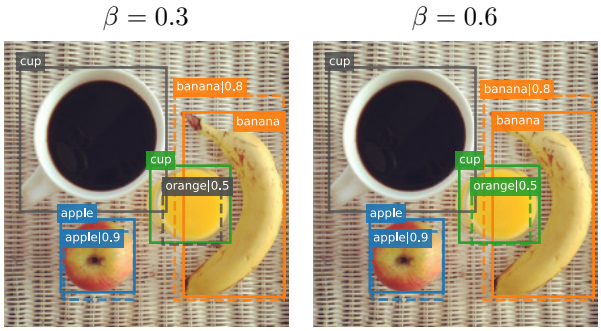

Figure 3. Assignment results with $\beta = 0.3$ and $\beta = 0.6$. A ground truth (solid line) and a detection (dashed line) with the same color are associated with each other. Gray represent that the ground truth or detections are not associated with any detections or ground truths. The parameter β controls the upper limit of the cost that a matched pair can take. Left: With smaller β, the detection incorrectly labeled as orange is considered as a false positive. Right: Larger β allow a detection with some errors to be associated with a ground truth.

The unit cost to transport to or from the dummy is a parameter β. The parameter β controls the level of an acceptable error for each assignment. With a smaller β, a detection is more likely to be assigned to the dummy because assigning a poor detection to a ground truth costs more than assigning it to a dummy.With a larger β, the requirement for an assignment is relaxed to some extent.When a downstream task requires precise detections, using a smaller β is recommended.

Figure 3 shows assignment results with different β. Associated detections and ground truths are presented with the same color. Gray indicates that the detection or the ground truth is assigned with the dummy. As in Fig. 3, with a smaller β, the detection with a classification error, which is labeled as orange, is assigned with the dummy, thus the detection is considered as a false positive. On the other hand, with a larger β, the classification error is allowed to some extent, and the detection is associated with a ground truth. We keep the parameters λ and β controllable so that developers can tune OC-cost for their problems.

3.3. Correction Cost Computation

Suppose a detector produces m detections and the image holds n ground truths. In our formulation, detections are represented by suppliers, and ground truths are represented by demanders. With the dummy detection, we have $m+1$ suppliers that hold s_i units where $i = 1, \ldots, m+1$. In the same way, the demanders need d_j units where $j = 1, \ldots, n+1$. We set the capacity of real suppliers $(s_1, \ldots, s_m)$ and demanders $(d_1, \ldots, d_n)$ to 1. The capacity of the dummy supplier s_{m+1} is set to n and the capacity of the dummy demander d_{n+1} is set to m. After comput-

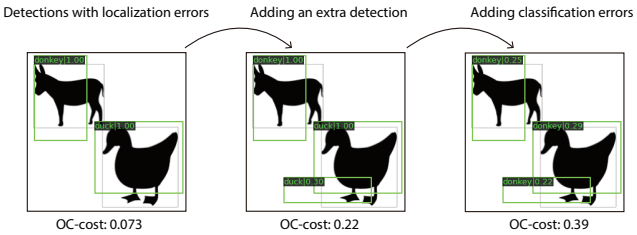

Figure 4. Different types of errors are sequentially added to detections. OC-cost monotonically increases at every step.

ing the correction cost matrix, we obtain the optimal assignment, *i.e.*, transportation plan π^*, by solving an optimal transportation problem. Based on the optimal assignment, we compute the correction cost. In this computation, the cost of dummy to dummy transportation is ignored, thus we set $\pi_{m+1,n+1}$ to 0. The transportation plan is normalized based on the remaining values.

$$\tilde{\pi}_{i,j} = \frac{\pi^*_{i,j}}{\sum_{i=1}^{m}\sum_{j=1}^{n}\pi^*_{i,j}}. \tag{8}$$

The final OC-cost $\tilde{c}$ is computed as:

$$\tilde{c} = \sum_{i=1}^{m+1}\sum_{j=1}^{n+1} c_{i,j}\tilde{\pi}_{i,j}. \tag{9}$$

Although the computational cost increases as the number of detections per image increases, the processing time remains reasonable for a practical number of detections. When the number of detections is equal to the number of ground truths, it takes about 22 seconds to compute OC-costs on the MS COCO validation set.

4. Experimental Results

Experiments are done on MS COCO validation 2017 split, which has 5000 images with annotated instances of 80 categories [14]. We test five off-the-shelf detectors: Faster-RCNN [18], RetinaNet [13], DETR [2], YOLOF [4], and VFNet [21]. All the detectors use the ResNet-50 backbone. We use pretained weights provided by MMDetection [3].

4.1. Toy Examples

To demonstrate how various types of errors affect OC-cost, we compute OC-cost on toy examples. Figure 4 shows three detection examples. From left to right, we sequentially add different types of detection errors to the ground truths. The left one shows the detections with small localization errors. The OC-cost remains low for the detections. In addition to the localization error, the middle one gets a false positive detection that increases the OC-cost by 0.147.

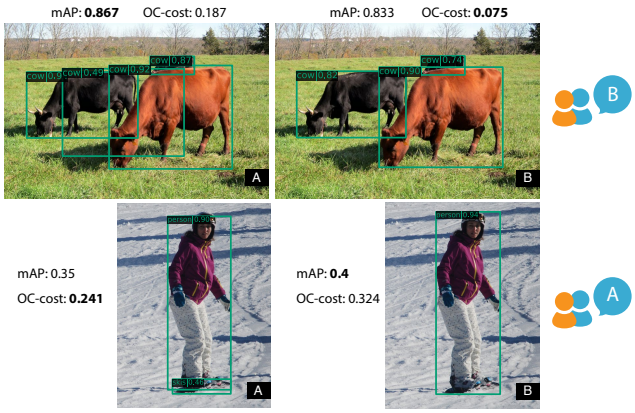

Figure 5. Examples of annotators' votes. Annotators compare two detection results and vote which one looks better. For each detection results, mAP for single image and OC-cost are displayed. OC-cost shows better agreement with human preference.

	OC-cost			mAP
(λ, β)	(0.2, 0.6)	(0.5, 0.6)	(0.5, 0.3)	
Accuracy	0.795	**0.806**	0.58	0.696

Table 1. Agreement with human preference. We compute OC-cost and mAP for a single image and compare the measures' preference to human's. The best accuracy is achieved by OC-cost at $\lambda = 0.5$ and $\beta = 0.6$.

In mAP, the effect of this false positive is determined depending on detections in other images, and thus this error may or may not affect mAP. In the right example, we perturb the predicted labels and confidence scores. This error also boosts the OC-cost. As these examples shows, OC-cost smoothly changes along with various types of image-level detection errors.

4.2. Agreement with Human Preference

We validate OC-cost's agreement with human preference. Annotators compare the results of two detectors and vote for the result which looks better. For each compared detections, we calculate OC-costs and check if the measure has agreement with the human annotators. We use RetinaNet and YOLOF as detectors, and sample 1057 images from the validation set. We choose the samples for which OC-cost with different hyper-parameters show different preferences. The aim of this sampling is to choose images where the two detectors differ to some extent, but the preference may not be too obvious. Three annotators worked for the data collection. We selected two out of three annotators with the highest Krippendorff's α [12], which represents the degree of agreement. Krippendorff's α for the selected annotators is 0.46. We omit samples where the

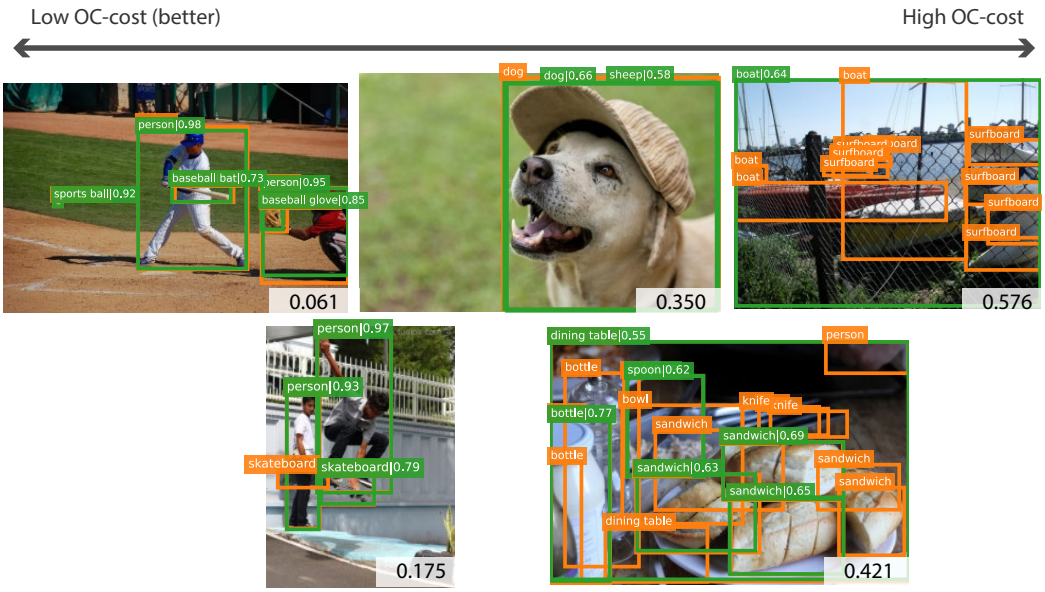

Figure 6. OC-cost examples. The parameters λ is 0.5, and β is 0.6. The detections (green) are produced by VFNet, and NMS is tuned on OC-cost. Ground truths are represented by orange bounding boxes. OC-cost is displayed on the right bottom of each image.

votes are split and obtain 772 samples. Figure 5 shows examples of the detection results, corresponding quantitative measures, and the annotators' votes.

We test different hyper-parameters λ and β for OC-cost by grid search. As a measure of agreement, accuracy is computed as a proportion of pairs which the automatic measure and human annotators select the same detection results. The results with three sets of parameters are shown in Table 1. The highest accuracy 0.80 is obtained at $\lambda = 0.5$ and $\beta = 0.6$. We also evaluate mAP for a single image. In a similar way to the standard mAP, we can obtain category-wise precision/recall curves for detections for a single image, and compute mAP. The accuracy of mAP for a single image is 0.69. This suggests that OC-cost is more consistent with human preference. In the following experiments, we use the best parameters $\lambda = 0.5$ and $\beta = 0.6$

4.3. Real-world Examples

To demonstrate OC-cost's behavior, we show detection examples on MS-COCO dataset and corresponding OC-costs in Fig. 6. From left to right, the examples are displayed in the order of OC-cost. The top left detection example, which successfully localizes and label the instances, obtains a fairly low cost. The top middle example correctly detects a dog but also has a redundant misclassified detection for which OC-cost penalizes. The top and bottom right examples got high OC-costs as they failed to detect many annotated instances.

	mAP (↑)	OC-cost (↓)
Faster-RCNN [18]	0.38	0.45
RetinaNet [13]	0.32	0.28
DETR [2]	**0.40**	0.57
YOLOF [4]	0.32	0.30
VFNet [21]	0.37	**0.26**

Table 2. mAP and OC-cost of the off-the-shelf detectors. mAP and OC-cost result in opposite detector's rankings. The result indicates that mAP and OC-cost evaluates different aspects of detectors.

	mAP (↑)	OC-Cost (↓)
Faseter-RCNN [18]	0.38	**0.45**
RetinaNet [13]	0.37	0.52
DETR [2]	0.40	0.57
YOLOF [4]	0.38	0.54
VFNet [21]	**0.44**	0.54

Table 3. Performance of the off-the-shelf detectors with NMS parameters tuned on mAP. The best detector is highlighted in bold.

4.4. Evaluating Detectors

We evaluate the off-the-shelf detectors in terms of mAP and OC-cost. Each detector's hyper-parameters for Non-maximum Suppression (NMS) are tuned on OC-cost. We does not conduct hyper-parameter tuning for DETR because it does not use NMS. Table 2 shows that the detectors' rank-

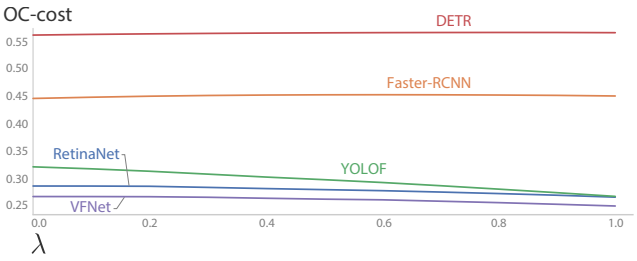

Figure 7. Detectors' OC-costs with different λ. With small λ, OC-cost emphasizes classification error, while larger λ emphasizes localization error. OC-costs of RetinaNet, YOLOF, and VFNet decrease with larger λ. This indicates that the error of the three detectors mainly come from classification errors

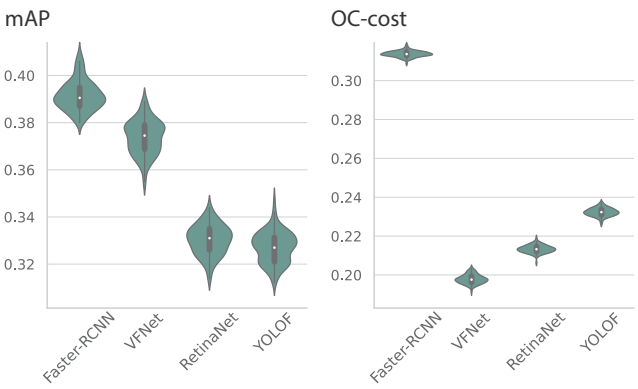

Figure 8. Distribution of mAP and OC-cost by bootstrapping with 100 trials. For each trial, we randomly sample 30% of the validation set and compute mAP and OC-cost. The distribution's overlaps across detectors imply that the detectors' rankings likely to flip by chance.

ings considerably differ on mAP and OC-cost. Note that this disagreement does not imply that one is correct and the other is not, because the measures evaluates different aspects of detections. mAP evaluates precision/recall curves of detected instances, whereas OC-cost evaluates the per image similarity of detections and ground truths.

OC-cost rates VFNet the best when NMS parameters are tuned on OC-cost. VFNet also performs the best in terms of mAP when NMS is tuned on mAP as shown in Tab. 3. This result indicates that VFNet successfully calibrates confidence scores and proposals, and we can filter out noisy detections easily by adjusting the confidence scores and the IOU threshold. DETR results in high OC-cost, because it outputs as many detections as possible, which leads to a high false positive rate. If the false positive detections are ranked lower, they hardly affect mAP. Different from mAP, OC-cost penalizes excessive number of detections.

Figure 7 shows the detectors' OC-costs with different λ. A smaller λ emphasizes classification costs, while a larger λ emphasizes localization costs. We observe that three detectors, RetinaNet, YOLOF, and VFNet decrease their OC-costs as λ gets larger. This indicates that their detection errors are mainly due to classification failures.

4.5. Consistency Analysis

A common practice of benchmarking detectors is to test detectors on a shared dataset and compare the evaluation measure. For reliable comparison, it is important to check that detectors rankings are not obtained by chance on a certain test set. To investigate the consistency of detectors' rankings on different test sets, we evaluate detectors on resampled datasets and check the distribution of the evaluation measures computed on those datasets. Specifically, for each trial, we randomly sample 30% of MS COCO validation 2017 split with replacement and calculate the measures on the resampled data. We repeat this process 100 times and report the distribution of sampled measurement values.

Figure 8 shows the distributions of OC-cost and mAP. The results of DETR is omitted for clarity. Full results are in the supplementary material. We observe that the distribution of mAP overlaps across detectors. This result indicates that the performance ranking can be flipped depending on the test set. On the other hand, OC-cost's variances remain low on smaller test sets, and the ranking of detectors is stable. This result suggests that OC-cost enables more reliable comparison of detectors.

4.6. Tuning NMS on OC-cost

As Tab. 2 shows, the preferences of mAP and OC-cost are very different. This suggests that we obtain completely different detectors depending on which measure is used to tune the detector. To clarify this, we compare two VFNet detectors whose NMS is tuned with mAP and OC-cost.

Figure 9 shows histograms of the number of detections per image. The mAP forces the detector to make more detections in order to increase the chances of a true positive detection. As a result, a detector tuned with mAP will make as many detections as possible in most images. On the other hand, OC-cost penalizes false positive detections, thus the number of detections per image is adjusted to be as many as ground truths. We can see that the distribution of the number of detections tuned with OC-cost is close to the ground truth's distribution. Figure 10 shows examples of detections tuned on mAP and OC-cost. OC-cost does not allow a large number of low confidence detections to be included for the purpose of increasing the recall.

mAP assumes that controlling the balance between precision and recall is each developer's responsibility. Thus, mAP may be useful when there is little knowledge about downstream tasks. We recommend to tune NMS over OC-

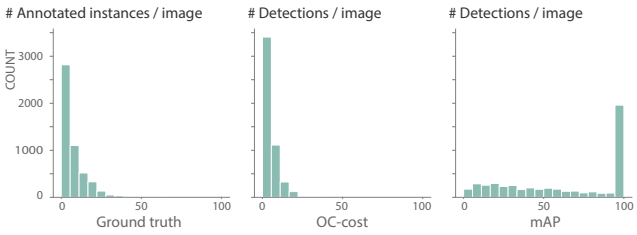

Figure 9. Distribution of number of detections by VFNet. Left: The number of annotated instances. Each image has 7.2 annotated instances on average. Center: When NMS parameters are tuned to minimize OC-cost, the number of detections are adjusted to be close to ground truths. Right: The detector tries to outputs as many detections as possible when NMS parameters are tuned with mAP.

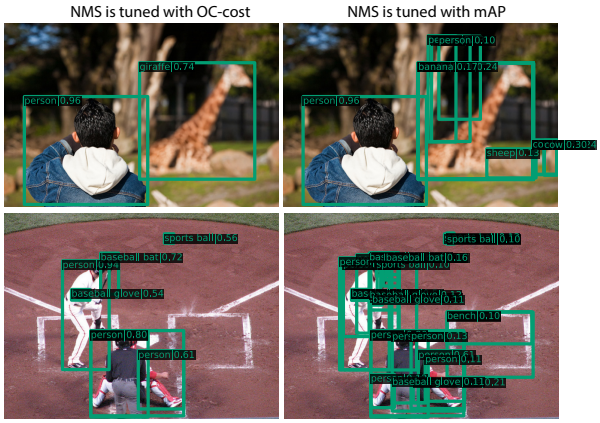

Figure 10. Detections with different NMS parameters. NMS parameters are tuned with OC-cost (Left) and mAP (Right). mAP encourages detectors to outputs many detections with low confidence scores because mAP's priority is to increase the chance to get true positives. Unlike mAP, OC-cost penalizes false positives.

cost instead of mAP when a detector needs to avoid redundant detections. OC-cost is particularly useful for applications that require detections to be filtered beforehand.

5. Discussions

Limitations. The main challenge in evaluation measures for object detection tasks is noise in annotations [10]. Object detection datasets inevitably contain noise in annotations. The noise can be introduced mainly due to ambiguity in categories/locations and annotators' skills. Like most evaluation measures, OC-cost also assumes that ground truths are correct. When ground-truth annotations have noise, OC-cost can underrate or overrate the detector's performance. Designing an evaluation measure considering a noise model representing how noise is generated will be an essential topic. Another issue is the category similarity for classification cost. Current OC-cost treats all categories equally. However, some applications have category-dependent risk of misclassification, *e.g.*, misclassifying a car into a truck is less critical than a person into a traffic light in autonomous driving. Considering category relationships is also an important issue of LVIS benchmark [9].

Social impact. Many real-world applications use object recognition technologies, and users of such applications are found in various regions, with different economic situations and cultural backgrounds. Object recognition technologies have to work well for all users, and if a technology disregards any group of users, such inequality has to be detected. A prior work reported that popular cloud services drop their performance for images from certain groups of users [6]. Per-image evaluation like OC-cost is useful to detect performance degradation in those cases.

The choice of the evaluation measure has a significant impact on the detector's behavior. This choice should be based on the various conditions of the final application. However, due to the complexity of such decision, it is possible to make a mistake in selection of a evaluation measure or hyper-parameters. Misuse of the evaluation measure can increase the risk of failure in the final application. To mitigate these problems, providing guidelines and typical application scenarios would be useful so that users understand the effects and limitations of using the measure intuitively.

6. Conclusions

We introduce a novel measure, OC-cost, for evaluating object detectors. We define a cost to correct detections to ground truths as a performance measure. The experimental results demonstrate that OC-cost is consistent with human preference to some extent. We also demonstrate that OC-cost has the capability to facilitate a fair comparison.

As we discussed, OC-cost and mAP evaluate detectors based on different assumptions. mAP focuses on evaluating ranking performance over a dataset, while OC-cost evaluates the per-image detection accuracy. For deeper understanding of detectors, we recommend using multiple evaluation measures that have different evaluation policies.

OC-cost can be considered as a dissimilarity measure for labeled bounding boxes. Not only for detector evaluation, extending OC-cost for other applications is an interesting future direction. A potential application is layout generation. OC-cost-like measures would be helpful in sampling or retrieving layouts [17] and analysis of generated layouts.

Acknowledgement. This work was partly supported by Academy of Finland project No. 324346, JST CREST Grant No. JPMJCR20D3 and FOREST Grant No. JPMJFR216O, Japan.

References

[1] Nicolas Bonneel, Michiel van de Panne, Sylvain Paris, and Wolfgang Heidrich. Displacement interpolation using lagrangian mass transport. *ACM Transactions on Graphics*, 30(6):1–12, 2011. 3

[2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 213–229, 2020. 5, 6

[3] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Xin Lu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. MMDetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019. 5

[4] Qiang Chen, Yingming Wang, Tong Yang, Xiangyu Zhang, Jian Cheng, and Jian Sun. You only look one-level feature. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13039–13048, 2021. 5, 6

[5] Achal Dave, Piotr Dollár, Deva Ramanan, Alexander Kirillov, and Ross Girshick. Evaluating large-vocabulary object detectors: The devil is in the details. *arXiv preprint arXiv:2102.01066*, 2021. 2

[6] Terrance de Vries, Ishan Misra, Changhan Wang, and Laurens van der Maaten. Does object recognition work for everyone? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2019. 8

[7] Mark Everingham, Luc Van Gool, Christopher K I Williams, John Winn, and Andrew Zisserman. The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010. 1, 2

[8] Zheng Ge, Songtao Liu, Zeming Li, Osamu Yoshie, and Jian Sun. OTA: Optimal transport assignment for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 303–312, 2021. 2, 3

[9] Agrim Gupta, Piotr Dollar, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3, 8

[10] David Hall, Feras Dayoub, John Skinner, Haoyang Zhang, Dimity Miller, Peter Corke, Gustavo Carneiro, Anelia Angelova, and Niko Suenderhauf. Probabilistic object detection: Definition and evaluation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1031–1040, 2020. 3, 8

[11] Derek Hoiem, Yodsawalai Chodpathumwan, and Qieyun Dai. Diagnosing error in object detectors. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 340–353, 2012. 3

[12] Krippendorff Klaus. Content analysis: An introduction to its methodology, 1980. 5

[13] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017. 5, 6

[14] 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. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 740–755, 2014. 2, 5

[15] Kemal Oksuz, Baris Can Cam, Emre Akbas, and Sinan Kalkan. Localization Recall Precision (LRP): A new performance metric for object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 521–537, 2018. 2

[16] Bahadır Özdemir, Selim Aksoy, Sandra Eckert, Martino Pesaresi, and Daniele Ehrlich. Performance measures for object detection evaluation. *Pattern Recognition Letters*, 31(10):1128–1137, 2010. 3

[17] Akshay Gadi Patil, Manyi Li, Matthew Fisher, Manolis Savva, and Hao Zhang. LayoutGMN: Neural graph matching for structural layout similarity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11048–11057, 2021. 8

[18] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (Neurips)*, pages 91–99, 2015. 5, 6

[19] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 658–666, 2019. 4

[20] Ellen M Voorhees and Donna Harman. Overview of TREC 2002. In *Trec*, 2002. 1

[21] Haoyang Zhang, Ying Wang, Feras Dayoub, and Niko Sunderhauf. Varifocalnet: An iou-aware dense object detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8514–8523, 2021. 5, 6

Optimal Correction Cost for Object Detection Evaluation

Supplementary Material

A. Details of the Annotation Experiment

The three annotators in Sec. 4.2 are employed as our in-house annotation team. We explained the purpose of the project to the annotators in advance, and they were able to ask any questions during the work. Each annotator completed annotating 1057 samples in four days. They reviewed paired detection results for a subset of the COCO Detection dataset [14] and assigned a binary preference to each pair. We did not store any personal information for this project. We believe that this annotation task does not violate the ethical principles in the CVPR ethics guidelines. We do not show the annotation interface in this supplementary material because it may reveal the authors' identity.

B. Full Results of Consistency Analysis

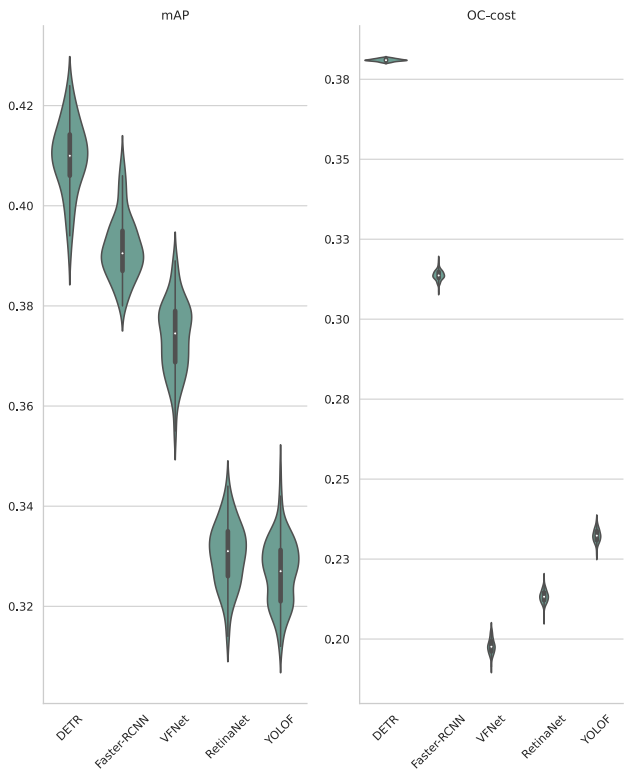

Figure 11. The full results of Fig. 8. Distributions of mAP and OC-cost are obtained by bootstrapping with 100 trials. The distributions' overlaps across detectors imply that the detectors' rankings likely to flip by chance.

We omit the DETR's result in Fig. 8 for visibility. We show full results in Fig. 11. The DETR's result does not change our conclusion that OC-cost's detectors' rankings are more stable than mAP.

C. Interactive Demo

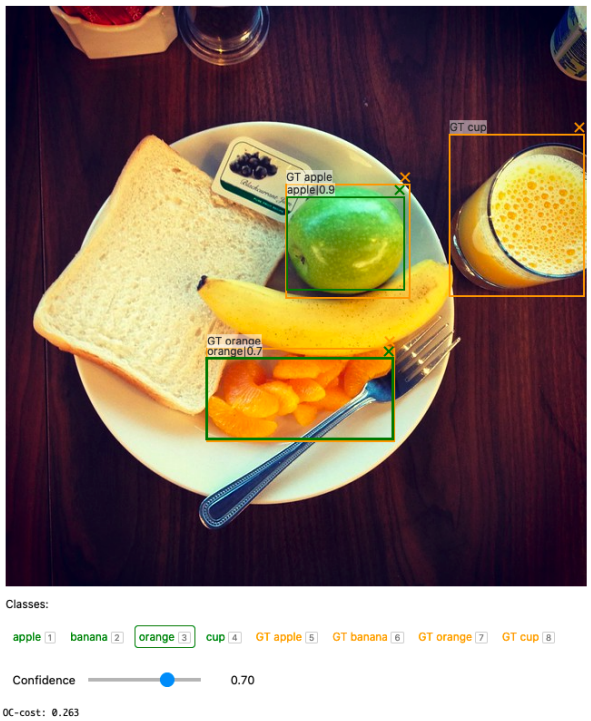

Figure 12. We can interactively give ground truths with orange boxes and detections with green ones. Once the ground truths and the detections are modified, corresponding OC-cost is displayed below the image.

We attach to this supplementary material a python notebook for an interactive demo. The screenshot of the demo is in Fig. 12. In the demo, OC-costs are computed for different detections and ground truths.

D. OC-cost Examples

We showcase detection examples on MS-COCO dataset and corresponding OC-costs in Fig. 13. From top to bottom, the examples are displayed in the order of OC-cost. The parameters λ is 0.5, and β is 0.6. The detections (green) are produced by VFNet, and NMS is tuned on OC-cost. Ground truths are represented by orange bounding boxes.

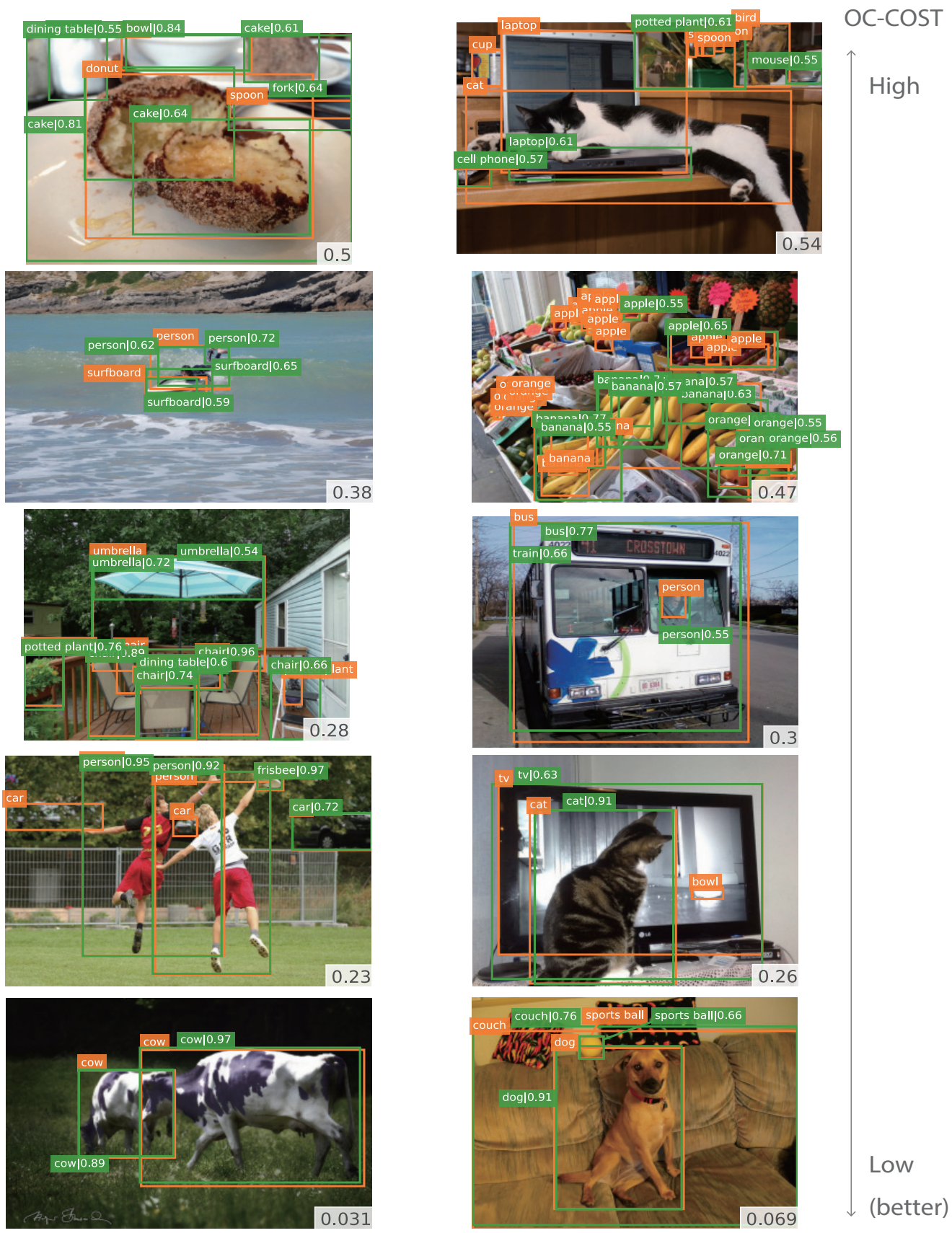

Figure 13. OC-cost examples. The parameters λ is 0.5, and β is 0.6. The detections (green) are produced by VFNet, and NMS is tuned on OC-cost. Ground truths are represented by orange bounding boxes. OC-cost is displayed on the right bottom of each image.