Object-Proposal Evaluation Protocol is 'Gameable'

Neelima Chavali*† Harsh Agrawal* Aroma Mahendru* Dhruv Batra
Virginia Tech
{gneelima, harsh92, maroma, dbatra}@vt.edu

Object proposals have quickly become the de-facto pre-processing step in a number of vision pipelines (for object detection, object discovery, and other tasks). Their performance is usually evaluated on partially annotated datasets. In this paper, we argue that the choice of using a partially annotated dataset for evaluation of object proposals is problematic – as we demonstrate via a thought experiment, the evaluation protocol is 'gameable', in the sense that progress under this protocol does not necessarily correspond to a "better" category independent object proposal algorithm.

To alleviate this problem, we: (1) Introduce a nearly-fully annotated version of PASCAL VOC dataset, which serves as a test-bed to check if object proposal techniques are overfitting to a particular list of categories. (2) Perform an exhaustive evaluation of object proposal methods on our introduced nearly-fully annotated PASCAL dataset and perform cross-dataset generalization experiments; and (3) Introduce a diagnostic experiment to detect the bias capacity *in an object proposal algorithm. This tool circumvents the need to collect a densely annotated dataset, which can be expensive and cumbersome to collect. Finally, we have released an easy-to-use toolbox which combines various publicly available implementations of object proposal algorithms which standardizes the proposal generation and evaluation so that new methods can be added and evaluated on different datasets. We hope that the results presented in the paper will motivate the community to test the category independence of various object proposal methods by carefully choosing the evaluation protocol.*

1. Introduction

In the last few years, the Computer Vision community has witnessed the emergence of a new class of techniques called *Object Proposal* algorithms [1–11].

Object proposals are a set of candidate regions or bounding boxes in an image that may potentially contain an object.

Object proposal algorithms have quickly become the de-facto pre-processing step in a number of vision pipelines – object detection [12–21], segmentation [22–26], object discovery [27–30], weakly supervised learning of object-object interactions [31, 32], content aware media retargeting [33], action recognition in still images [34] and visual tracking [35, 36]. Of all these tasks, object proposals have been particularly successful in object detection systems. For example, *nearly all top-performing entries* [13, 37–39] in the ImageNet Detection Challenge 2014 [40] used object proposals. They are preferred over the formerly used sliding window paradigm due to their computational efficiency. Objects present in an image may vary in location, size, and aspect ratio. Performing an exhaustive search over such a high dimensional space is difficult. By using object proposals, computational effort can be focused on a small number of candidate windows.

The focus of this paper is the protocol used for evaluating object proposals. Let us begin by asking – *what is the purpose of an object proposal algorithm?*

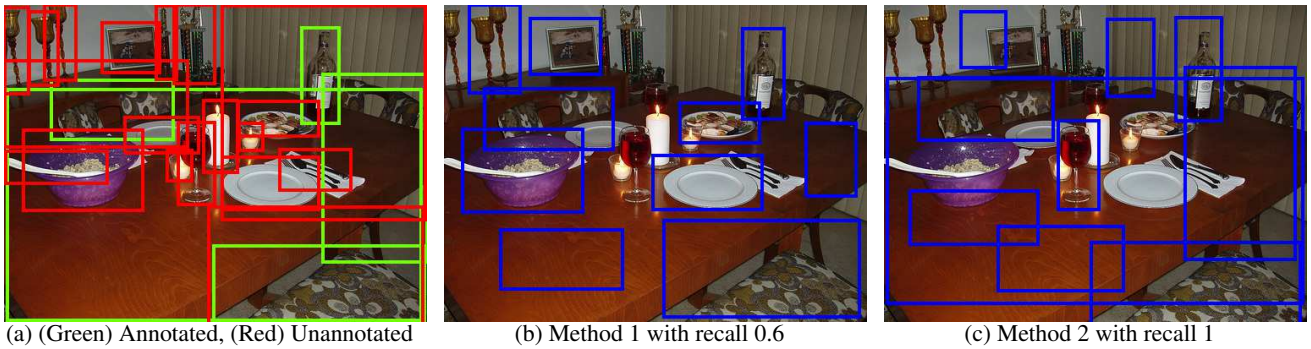
In early works [2, 4, 6], the emphasis was on *category independent object proposals*, where the goal is to identify instances of *all* objects in the image irrespective of their category. While it can be tricky to precisely define what an "object" is[1], these early works presented cross-category evaluations to establish and measure category independence.

More recently, object proposals are increasingly viewed as *detection proposals* [1, 8, 11, 42] where the goal is to improve the object detection pipeline, focusing on a chosen set of object classes (*e.g.* ~20 PASCAL categories). In fact, many modern proposal methods are learning-based [9–11, 42–46] where the definition of an "object" is the set of annotated classes in the dataset. This increasingly blurs the boundary between a proposal algorithm and a detector.

Notice that the former definition has an emphasis on object discovery [27, 28, 30], while the latter definition emphasises on the ultimate performance of a detection pipeline. Surprisingly, despite the two different goals of 'object pro-

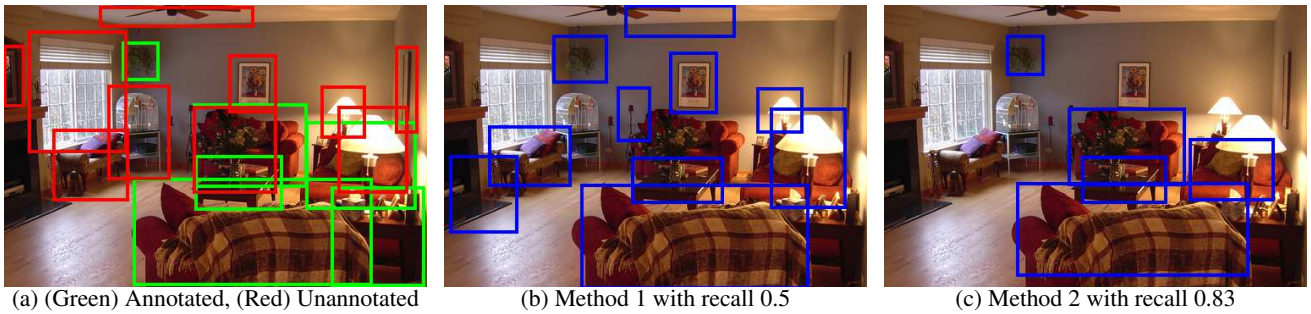
*Equal contribution.
†Now at Amgen Inc.

[1]Most category independent object proposal methods define an object as "stand-alone thing with a well-defined closed-boundary". For "thing" *vs.* "stuff" discussion, see [41].



(a) (Green) Annotated, (Red) Unannotated (b) Method 1 with recall 0.6 (c) Method 2 with recall 1

Figure 1: (a) shows PASCAL annotations natively present in the dataset in green. Other objects that are not annotated but present in the image are shown in red; (b) shows Method 1 and (c) shows Method 2. Method 1 visually seems to recall more categories such as plates, glasses, *etc*. that Method 2 missed. Despite that, the computed recall for Method 2 is higher because it recalled all instances of PASCAL categories that were present in the ground truth. Note that the number of proposals generated by both methods is equal in this figure.



(a) (Green) Annotated, (Red) Unannotated (b) Method 1 with recall 0.5 (c) Method 2 with recall 0.83

Figure 2: (a) shows PASCAL annotations natively present in the dataset in green. Other objects that are not annotated but present in the image are shown in red; (b) shows Method 1 and (c) shows Method 2. Method 1 visually seems to recall more categories such as lamps, picture, *etc*. that Method 2 missed. Clearly the recall for Method 1 *should* be higher. However, the calculated recall for Method 2 is significantly higher, which is counter-intuitive. This is because Method 2 recalls more PASCAL category objects.

posal,' there exists only a single evaluation protocol:

1. Generate proposals on a dataset: The most commonly used dataset for evaluation today is the PASCAL VOC [47] detection set. Note that this is a *partially annotated* dataset where only the 20 PASCAL category instances are annotated.
2. Measure the performance of the generated proposals: typically in terms of 'recall' of the annotated instances. Commonly used metrics are described in Section 3.

The central thesis of this paper is that the current evaluation protocol for object proposal methods is suitable for object detection pipeline but is a *'gameable' and misleading protocol* for category independent tasks. By evaluating only on a specific set of object categories, we fail to capture the performance of the proposal algorithms on *all the remaining object categories that are present in the test set, but not annotated in the ground truth*.

Figs. 1, 2 illustrate this idea on images from PASCAL VOC 2010. Column (a) shows the ground-truth object annotations (in green, the annotations natively present in the dataset for the 20 PASCAL categories –'chairs', 'tables', 'bottles', *etc*.; in red, the annotations that we added to the dataset by marking object such as 'ceiling fan', 'table lamp', 'window', *etc*. originally annotated 'background' in the dataset). Columns (b) and (c) show the outputs of two object proposal methods. Top row shows the case when both methods produce the same number of proposals; bottom row shows unequal number of proposals. We can see that proposal method in Column (b) seems to be more "complete", in the sense that it recalls or discovers a large number of instances. For instance, in the top row it detects a number of non-PASCAL categories ('plate', 'bowl', 'picture frame', *etc*.) but misses out on finding the PASCAL category 'table'. In both rows, the method in Column (c) is reported as achieving a higher recall, *even in the bottom row, when it recalls strictly fewer objects, not just different ones*. The reason is that Column (c) recalls/discovers instances of the 20 PASCAL categories, which are the only ones annotated in the dataset. Thus, Method 2 appears to be a *better* object proposal generator simply because it focuses on the annotated categories in the dataset.

While intuitive (and somewhat obvious) in hindsight, we believe this is a crucial finding because it makes the current protocol *'gameable'* or susceptible to manipulation (both intentional and unintentional) and misleading for measuring improvement in category independent object proposals.

Some might argue that if the end task is to detect a cer-

tain set of categories (20 PASCAL or 80 COCO categories) then it is enough to evaluate on them and there is no need to care about other categories which are not annotated in the dataset. We agree, but it is important to keep in mind that object detection is not the only application of object proposals. There are other tasks for which it is important for proposal methods to generate category independent proposals. For example, in semi/unsupervised object localization [27–30] the goal is to identify all the objects in a given image that contains many object classes without any specific target classes. In this problem, there are no image-level annotations, an assumption of a single dominant class, or even a known number of object classes [28]. Thus, in such a setting, using a proposal method that has tuned itself to 20 PASCAL objects would not be ideal – in the worst case, we may not discover any new objects. As mentioned earlier, there are many such scenarios including learning object-object interactions [31, 32], content aware media retargeting [33], visual tracking [36], *etc*.

To summarize, the contributions of this paper are:

- We report the 'gameability' of the current object proposal evaluation protocol.
- We demonstrate this 'gameability' via a simple thought experiment where we propose a 'fraudulent' object proposal method that *significantly outperforms all existing object proposal techniques* on current metrics, but would under any no circumstances be considered a category independent proposal technique. As a side contribution of our work, we present a simple technique for producing state-of-art object proposals.
- After establishing the problem, we propose three ways of improving the current evaluation protocol to measure the category independence of object proposals:
  1. evaluation on *fully* annotated datasets,
  2. cross-dataset evaluation on *densely* annotated datasets.
  3. a new evaluation metric that quantifies the *bias capacity* of proposal generators.

  For the first test, we introduce a nearly-fully annotated PASCAL VOC 2010 where we annotated *all instances of all object categories* occurring in the images.
- We thoroughly evaluate existing proposal methods on this nearly-fully and two densely annotated datasets.
- We have released release all code and data for experiments[2], and an object proposals library that allows for comparison of popular object proposal techniques.

2. Related Work

Types of Object Proposals: Object proposals can be broadly categorized into two categories:

- **Window scoring**: In these methods, the space of all possible windows in an image is sampled to get a subset of the windows (*e.g.*, via sliding window). These windows are then scored for the presence of an object based on the image features from the windows. The algorithms that fall under this category are [1, 4, 5, 10, 45, 48].
- **Segment based**: These algorithms involve over-segmenting an image and merging the segments using some strategy. These methods include [2, 3, 6–9, 11, 44, 46, 49]. The generated region proposals can be converted to bounding boxes if needed.

Beyond RGB proposals: Beyond the ones listed above, a wide variety of algorithms fall under the umbrella of 'object proposals'. For instance, [50–54] used spatio-temporal object proposals for action recognition, segmentation and tracking in videos. Another direction of work [55–57] explores use of RGB-D cuboid proposals in an object detection and semantic segmentation in RGB-D images. While the scope of this paper is limited to proposals in RGB images, the central thesis of the paper (*i.e.*, gameability of the evaluation protocol) is broadly applicable to other settings.

Evaluating Proposals: There has been a relatively limited analysis and evaluation of proposal methods or the proposal evaluation protocol. Hosang *et al*. [58] focus on evaluation of object proposal algorithms, in particular the stability of such algorithms on parameter changes and image perturbations. Their works shows that a large number of category independent proposal algorithms indeed generalize well to non-PASCAL categories, for instance in the ImageNet 200 category detection dataset [40]. Although these findings are important (and consistent with our experiments), they are unrelated to the 'gameability' of the evaluation protocol. In [59], authors present an analysis of various proposal methods regarding proposal repeatability, ground truth annotation recall, and their impact on detection performance. They also introduced a new evaluation metric (Average Recall). Their argument for a new metric is the need for a better localization between generated proposals and ground truth. While this is a valid and significant concern, it is orthogonal to the'gameability' of the evaluation protocol, which to the best of our knowledge has not been previously addressed. Another recent related work is [60], which analyzes various methods in segment-based object proposals, focusing on the challenges faced when going from PASCAL VOC to MS COCO. They also analyze how aligned the proposal methods are with the bias observed in MS COCO towards small objects and the center of the image and propose a method to boost their performance. Although there is a discussion about biases in datasets but it is unlike our theme, which is 'gameability' due to these biases. As stated earlier, while early papers [2,4,6] reported cross-dataset or cross-category generalization experiments similar to ones reported in this paper, with the trend of learning-based proposal methods, these experiments and concerns seem to have fallen out of standard practice, which we show is problematic.

[2]Data and code can be accessed at: https://filebox.ece.vt.edu/~aroma/web/object-proposals.html

3. Evaluating Object Proposals

Before we describe our evaluation and analysis, let us first look at the object proposal evaluation protocol that is widely used today. The following two factors are involved:

1. **Evaluation Metric**: The metrics used for evaluating object proposals are all typically functions of intersection over union (IOU) (or Jaccard Index) between generated proposals and ground-truth annotations. For two boxes/regions b_i and b_j, IOU is defined as:

$$\text{IOU}(b_i, b_j) = \frac{area(b_i \cap b_j)}{area(b_i \cup b_j)} \quad (1)$$

The following metrics are commonly used:

• **Recall @ IOU Threshold** t: For each ground-truth instance, this metric checks whether the 'best' proposal from list L has IOU greater than a threshold t. If so, this ground truth instance is considered 'detected' or 'recalled'. Then average recall is measured over all the ground truth instances:

$$\text{Recall} @ t = \frac{1}{|G|} \sum_{g_i \in G} \text{I}\left[\max_{l_j \in L} \text{IOU}(g_i, l_j) > t\right], \quad (2)$$

where $I[\cdot]$ is an indicator function for the logical preposition in the argument. Object proposals are evaluated using this metric in two ways:

– plotting Recall-*vs*.-#proposals by fixing t

– plotting Recall-*vs*.-t by fixing the #proposals in L.

• **Area Under the recall Curve (AUC)**: AUC summarizes the area under the Recall-*vs*.-#proposals plot for different values of t in a single plot. This metric measures AUC-*vs*.-#proposals. It is also plotted by varying #proposals in L and plotting AUC-vs-t.

• **Volume Under Surface (VUS)**: This measures the average recall by linearly varying t and varying the #proposals in L on either linear or log scale. Thus it merges both kinds of AUC plots into one.

• **Average Best Overlap (ABO)**: This metric eliminates the need for a threshold. We first calculate the overlap between each ground truth annotation $g_i \in G$, and the 'best' object hypotheses in L. ABO is calculated as the average:

$$\text{ABO} = \frac{1}{|G|} \sum_{g_i \in G} \max_{l_j \in L} \text{IOU}(g_i, l_j) \quad (3)$$

ABO is typically is calculated on a per class basis. Mean Average Best Overlap (MABO) is defined as the mean ABO over all classes.

• **Average Recall (AR)**: This metric was recently introduced in [59]. Here, average recall (for IOU between 0.5 to 1)-*vs*.-#proposals in L is plotted. AR also summarizes proposal performance across different values of t. AR was shown to correlate with ultimate detection performance better than other metrics.

2. **Dataset**: The most commonly used datasets are the PASCAL VOC [47] detection datasets. Note that these are *partially annotated* datasets where only the 20 PASCAL category instances are annotated. Recently analyses have been shown on ImageNet [58], which has more categories annotated than PASCAL, but is still a partially annotated dataset.

4. A Thought Experiment: How to Game the Evaluation Protocol

Let us conduct a thought experiment to demonstrate that the object proposal evaluation protocol can be 'gamed'.

Imagine yourself reviewing a paper claiming to introduce a new object proposal method – called DMP.

Before we divulge the details of DMP, consider the performance of DMP shown in Fig. 3 on the PASCAL VOC 2010 dataset, under the AUC-*vs*.-#proposals metric.

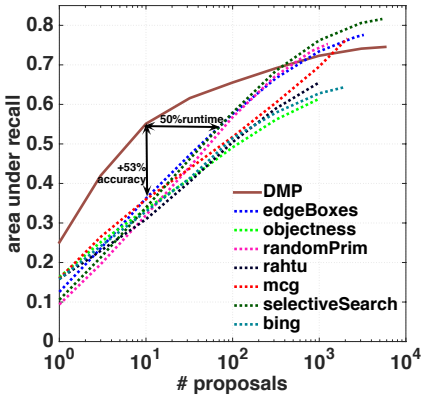


Figure 3: Performance of different object proposal methods (dashed lines) and our proposed 'fraudulent' method (DMP) on the PASCAL VOC 2010 dataset. We can see that DMP *significantly* outperforms all other proposal generators. See text for details.

As we can clearly see, the proposed method DMP *significantly* exceeds all existing proposal methods [1–6, 8, 10, 11] (which seem to have little variation over one another). The improvement at some points in the curve (*e.g.*, at M=10) seems to be *an order of magnitude* larger than all previous incremental improvements reported in the literature! In addition to the gain in AUC at a fixed M, DMPs also achieves the same AUC (0.55) at an *order of magnitude fewer* number of proposals (M=10 *vs*. M= 50 for edgeBoxes [1]). Thus, fewer proposals need to be processed by the ensuing detection system, resulting in an equivalent run-time speedup. This seems to indicate that a significant progress has been made in the field of generating object proposals.

So what is our proposed state-of-art technique DMP?
It is a mixture-of-experts model, consisting of 20 experts, where each expert is a deep feature (fc7)-based [61] objectness detector. At this point, you, the savvy reader, are probably already beginning to guess what we did.