UniT: Unified Knowledge Transfer for Any-shot Object Detection and Segmentation

Siddhesh Khandelwal[*,1,2] Raghav Goyal[*,1,2] Leonid Sigal[1,2,3]
[1]Department of Computer Science, University of British Columbia
[2]Vector Institute for AI [3]CIFAR AI Chair
skhandel@cs.ubc.ca rgoyal14@cs.ubc.ca lsigal@cs.ubc.ca

Abstract

Methods for object detection and segmentation rely on large scale instance-level annotations for training, which are difficult and time-consuming to collect. Efforts to alleviate this look at varying degrees and quality of supervision. Weakly-supervised approaches draw on image-level labels to build detectors/segmentors, while zero/few-shot methods assume abundant instance-level data for a set of base *classes, and none to a few examples for* novel *classes. This taxonomy has largely siloed algorithmic designs. In this work, we aim to bridge this divide by proposing an intuitive and unified semi-supervised model that is applicable to a range of supervision: from zero to a few instance-level samples per* novel *class. For* base *classes, our model learns a mapping from weakly-supervised to fully-supervised detectors/segmentors. By learning and leveraging visual and lingual similarities between the* novel *and* base *classes, we transfer those mappings to obtain detectors/segmentors for* novel *classes; refining them with a few* novel *class instance-level annotated samples, if available. The overall model is end-to-end trainable and highly flexible*[1]*. Through extensive experiments on MS-COCO [32] and Pascal VOC [14] benchmark datasets we show improved performance in a variety of settings.*

1. Introduction

Over the past decade CNNs have emerged as the dominant building blocks for various computer vision understanding tasks, including object classification [21, 45, 52], detection [33, 42, 43], and segmentation [8, 20]. Architectures based on Faster R-CNN [43], Mask R-CNN [20] and YOLO [42] have achieved impressive performance on a variety of core vision tasks. However, traditional CNN-based approaches rely on lots of supervised data for which the annotation efforts can be time-consuming and expensive [22, 29]. While image-level class labels are easy to obtain, more structured labels such as bounding boxes or segmentations are difficult and expensive[2]. Further, in certain domains (*e.g.*, medical imaging) more detailed labels may require subject expertise. The growing need for efficient learning has motivated development of various approaches and research sub-communities.

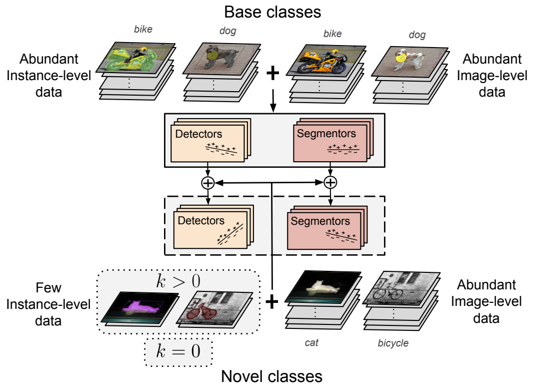

Figure 1: **Semi-supervised Any-shot Detection and Segmentation.** The data used in our setting is categorized in two ways: (1) image-level classification data for *all* object classes, and (2) abundant instance data for *base* object classes and limited (possibly zero) instance data for *novel* object classes, with the aim to obtain a model that learns to detect/segment both base and novel objects at test time.

On one end of the spectrum, *zero-shot learning* methods require no visual data and use auxiliary information, such as attributes or class names, to form detectors for unseen classes from related seen category detectors [3, 16, 40, 65]. *Weakly-supervised learning* methods [2, 5, 12, 29, 34, 61] aim to utilize readily available coarse image-level labels for more granular downstream tasks, such as object detection [3, 40] and segmentation [29, 71]. Most recently, *few-shot learning* [1, 41, 49, 60] has emerged as a learning-to-learn paradigm which either learns from few labels directly or by simulation of few-shot learning paradigm through meta-learning [15, 47, 57]. An interesting class of *semi-supervised* methods [17, 22, 26, 56, 58, 68] have emerged which aim

[1] Code is available at https://github.com/ubc-vision/UniT
[*] Denotes equal contribution

[2] Segmentation annotations in PASCAL VOC take 239.7 seconds/image, on average, as compared to 20 seconds/image for image-level labels [4].

to transfer knowledge from abundant *base* classes to data-starved *novel* classes, especially for granular instance-level visual understanding tasks. However, to date, there isn't a single, unified framework that can effectively leverage various forms and amounts of training data (zero-shot to fully supervised).

We make two fundamental observations that motivate our work. First, image-level supervision is abundant, while instance-level structured labels, such as bounding boxes and segmentation masks, are expensive and scarce. This is reflected in the scales of widely used datasets where classification tasks have $>$ 5K classes [28, 52] while the popular object detection/segmentation datasets, like MSCOCO [32], have annotations for only 80 classes. A similar observation was initially made by Hoffman *et al.* [22] and other semi-supervised [26, 56, 58] approaches. Second, the assumption of no instance-level supervision for target classes (as is the case for semi-supervised [22, 26, 56, 58] and zero-shot methods [3, 16, 40, 65]) is artificial. In practice, it is often easy to collect few instance-level annotations and, in general, a good object detection/segmentation model should be robust and work with *any* amount of available instance-level supervision. Our motivation is to bridge weakly-supervised, zero- and few-shot learning paradigms to build an expressive, simple, and interpretable model that can operate across types (weak/strong) and amounts of instance-level supervision (from 0 to 90+ instance-level samples per class).

We develop a unified semi-supervised framework (UniT) for object detection and segmentation that scales with different levels of instance-level supervision (see Figure 1). The data used in training our model is categorized in two ways, (1) image-level classification data for *all* the object classes, and (2) abundant detection data for a set of *base* object classes and limited (possibly zero) detection data for a set of *novel* object classes, with the aim to obtain a model that learns to detect both *base* and *novel* objects at test time.

Our algorithm, illustrated in Figure 2, jointly learns weak-detectors for *all* the object classes, from image-level classification data, and supervised regressors/segmentors on top of those for *base* classes (based on instance-level annotations in a supervised manner). The classifiers, regressors and segmentors of the *novel* classes are expressed as a weighted linear combination of its base class counterparts. The weights of the combination are determined by a multi-modal similarity measure: *lingual* and *visual*. The key insight of our approach is to utilize the multi-modal similarity measure between the novel and base classes to enable effective knowledge transfer and adaptation. The adopted *novel* classifier/regressors/segmentors can further be refined based on instance-level supervision, if any available. We experiment with the widely-used detection/segmentation datasets - Pascal VOC [13] and MSCOCO [32], and compare our method with state-of-the-art few-shot, weakly-supervised, and semi-supervised object detection/segmentation methods.

Contributions: Our contributions can be summarized as follows: (1) We study the problem of semi-supervised object detection and segmentation in light of image-level supervision and limited instance-level annotations, ranging from no data (zero-shot) to a few (few-shot); (2) We propose a general, unified, interpretable, and flexible end-to-end framework that, by leveraging a learned multi-modal (lingual + visual) similarity metric, can adopt classifiers/detectors/segmentors for *novel* classes by expressing them as linear combinations of their *base* class counterparts. (3) In the context of our model, we contrast the relative importance of weak image-level supervision with strong instance-level supervision, and highlight the importance of the former under a small fixed annotation budget (4) We illustrate the flexibility and effectiveness of our model by applying it to a variety of tasks (object detection and segmentation) and datasets (Pascal VOC [13], MSCOCO [32]); showing state-of-the-art performance. We get up to 23% relative improvement in mAP over the closest semi-supervised methods [17], and up to 16% gain over the best performing few-shot method [62] under a fixed annotation budget. We conduct comprehensive comparisons across settings, tasks, types and levels of supervision.

2. Related Work

Few-shot object detection: Object detection with limited data was initially explored in a transfer learning setting by Chen *et al.* [7]. In the context of meta-learning [1, 15, 41, 49, 60], Kang *et al.* [24] developed a few-shot model where the learning procedure is divided into two phases: first the model is trained on a set of *base* classes with abundant data using episodic tasks, then, in the second phase, a few examples of *novel* classes and *base* classes are used for fine tuning the model. Following this formulation, [63, 67] employed better performing architecture - Faster R-CNN [43], instead of a one-stage YOLOv2 [42]. Yan *et al.* [67] extended the problem formulation to account for segmentation in addition to detection. In contrast to the above approaches, Wang *et al.* [62] showed that meta-learning is not a crucial ingredient to Few-shot object detection, and simple fine-tuning produces better detectors. Similar to the above works, we also adopt the two-phase learning procedure. However, we fundamentally differ in assuming that easily attainable extra supervision, in the form of image-level data, over *all* the classes is available. Unlike [63], we learn a semantic mapping between *weakly-supervised detectors* and detectors obtained using a large number of examples.

Weakly-supervised object detection: Weak supervision in object detection takes the form of image-level labels, usually coupled with bounding box proposals [59, 73], thereby representing each image as a bag of instances [2, 5, 9, 12, 18, 34, 44, 50, 54, 55, 61, 70]. Bilen *et al.* [5] proposed an end-to-end architecture which softly labeled object proposals and uses a detection stream, in addition to classification stream, to classify them. Further extensions followed, Diba *et al*. [12] incorporated better proposals into a cascaded deep network; Tang *et al*. [55] proposed an Online Instance Classifier Refinement (OICR) algorithm which iteratively refines predictions. More recently, further improvements were made by combining weakly-supervised learning with strongly-supervised detectors, by treating predicted locations from the weakly-supervised detector as pseudo-labels for a strongly-supervised variant [2, 61]. In this work, we choose to adopt and build on top of single-stage OICR [55], hence enabling end-to-end training. However, our approach is not limited to the choice of weakly-supervised architecture.

Semi-supervised object detection: Approaches under semi-supervised setup assume abundant detection data for *base* classes and no detection data for *novel* classes, in addition to weak supervision for *all* the classes. The methods in this category first learn weak classifiers for *all* classes using abundant weak supervision, then fine-tune *base* classifiers into detectors using abundant detection data, and finally transfer this transformation to obtain detectors for *novel* classes using an external (or learned) similarity measure between *base* and *novel* classes. LSDA [22], being the first, formed similarity based on L2-normalized weak classifier weights. Tang *et al*. [56] extended this approach to include semantic and visual similarity explicitly. DOCK [26] expanded the types of similarities to include spatial and attribute cues using external knowledge sources. Other works leverage semantic hierarchies of classes, such as Yang *et al*. [68] proposes a class split based on granularity of classes, and transfers knowledge from coarse to fine grained classes. Uijlings *et al*. [58] uses a proposal generator trained on base classes, and transfers the proposals from base to novel classes by computing their similarity on a tree based on Imagenet semantic hierarchy [45]. Similar to the above methods we also use visual and lingual similarities between base and novel classes, but consider a more general problem setting where we have varying degrees of detection supervision for novel classes ranging from zero to a few k-samples per class.

Unique, and closest to our setup, is NOTE-RCNN [17]. In [17], few-k detection samples for *novel* classes are used as seed annotations, based on which training-mining [55, 58] is employed. Specifically, they initialize detectors for *novel* classes by training them with few seed annotations, and iteratively refine them by retraining with mined bounding boxes for novel classes. They transfer knowledge indirectly in the form of losses that act as regularizers. Our approach, on the other hand, takes on a simpler and more intuitive direction where we first transfer the mappings from *base* to *novel* classes, and use few seed annotations (if available) to fine-tune the detectors. Despite being simpler, our approach is more accurate, and works in the $k = 0$ regime. Further, unlike all the above semi-supervised approaches , we transfer across tasks, including regression and segmentation.

Zero-shot object detection: Zero-shot approaches rely on auxiliary semantic information to connect *base* and *novel* classes; *e.g.*, text description of object labels or their attributes [3, 16, 40, 65]. A common strategy is to represent *all* classes as prototypes in the semantic embedding space and to learn a mapping from visual features to this embedding space using *base* class data; classification is then obtained using nearest distance to *novel* prototypes. This approach was expended to detection in [10, 27, 30, 46, 69, 72]. Bansal *et al*. [3], similarly, proposed method to deal with situations where objects from novel/unseen classes are present in the background regions. We too explore the setting where we are not provided with any instance data for novel classes, but in addition assume weak-supervision for novel object classes in the form of readily available [28] image-level annotations.

3. Problem Formulation

Here we formally introduce the semi-supervised any-shot object detection / segmentation setup. We start by assuming image-level supervision for *all* the classes denoted by $\mathcal{D}^{class} = \{(\mathbf{x}_i, \mathbf{a}_i)\}$, where each image $\mathbf{x}_i$ is annotated with a label $\mathbf{a}_i \in \{0,1\}^{|\mathcal{C}|}$, where $a_i^j = 1$ if image $\mathbf{x}_i$ contains at least one j-th object, indicating its presence; $\mathbf{a}_i = \{a_i^j\}_{j=1}^{|\mathcal{C}|}$ with $|\mathcal{C}|$ being number of object classes.

We further extend the above image-level data with object-instance annotations by following the few-shot object detection formulation [24, 63, 67]. We split the classes into two disjoint sets: *base* classes $\mathcal{C}_{base}$ and *novel* classes $\mathcal{C}_{novel}$; $\mathcal{C}_{base} \cap \mathcal{C}_{novel} = \emptyset$. For base classes, we have abundant instance data $\mathcal{D}_{base} = \{(\mathbf{x}_i, \mathbf{c}_i, \mathbf{y}_i)\}$, where $\mathbf{x}_i$ is an input image, $\mathbf{c}_i = \{c_{i,j}\}$ are class labels, $\mathbf{y}_i = \{\mathbf{bbox}_{i,j}\}$ or $\mathbf{y}_i = \{\mathbf{s}_{i,j}\}$ are corresponding bounding boxes and/or masks for each instance j in image i. For *novel* classes, we have limited instance data $\mathcal{D}_{novel} = \{(\mathbf{x}_i, \mathbf{c}_i, \mathbf{y}_i)\}_{i=1,\ldots,k}$, where data for k-shot detection / segmentation only has k bounding boxes / masks for each novel class in $\mathcal{C}_{novel}$. Note, these annotations are assumed only for images in the *train* data. Also, for semi-supervised zero-shot, $k = 0$ and $\mathcal{D}_{novel} = \emptyset$.

4. Approach

We propose a single unified framework that leverages the weak image-level supervision for object detection / segmentation in any-shot setting. That is, our proposed approach can seamlessly incorporate arbitrary levels of instance-level supervision without the need to alter the architecture.

Our proposed framework builds upon the Faster R-CNN [43] / Mask R-CNN [20] architecture. Faster R-CNN [43] utilizes a two-stage pipeline in order to perform object detection. The first stage uses a region proposal network (RPN) to generate class-agnostic object region proposals $\{\mathbf{rbox}_{i,j}\}$ for image i. The second stage is a detection network (Fast R-CNN [19]) that performs RoI pooling, forming feature vector $\mathbf{z}_{i,j} = \texttt{RoIAlign}(\mathbf{x}_i, \mathbf{rbox}_{i,j})$ for proposal j in im-

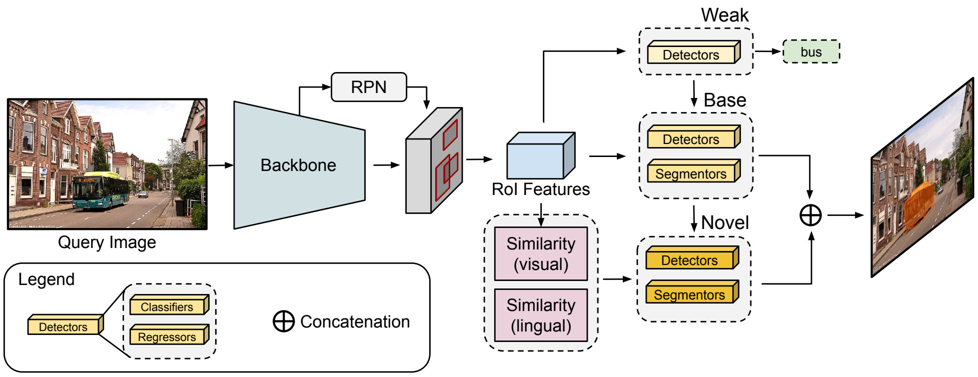

Figure 2: **Overall Architecture.** We form detectors/segmentors for *base* classes as a refinement on top of weak detectors. The detectors/segmentors for *novel* classes utilize a similarity weighed transfer (pink boxes) from the base class refinements. In a k-shot setting, (few) novel class instance annotations are incorporated through direct adaptation of the resulting *novel* detectors/segmentors through fine-tuning. All detectors are built on top of Faster/Mask RCNN architecture which comprises of classification and regression heads with shared backbone (in cyan) and simultaneously trained region proposal network (RPN).

age i, and learns to classify this RoI feature vector $\mathbf{z}$ (we drop proposal and image indexing for brevity for remainder of the section) into one of the object classes and refine the bounding box proposals using a class-aware regressors. Conceptually, an R-CNN object detector can be thought of as a combination of a classifier and regressor (see Figure 2). Mask R-CNN [20] is a simple extension to the Faster R-CNN framework, wherein an additional head is utilized in the second stage to predict the instance segmentation masks.

Figure 2 details the proposed architecture. The model consists of two branches: i) the weakly-supervised branch that trains detectors $\hat{c} = \texttt{softmax}(f_{\mathbf{W}^{weak}}(\mathbf{z}))$ using image-level supervision $\mathcal{D}^{class}$, and ii) a supervised branch that uses detection data $\mathcal{D}_{base}/\mathcal{D}_{novel}$ to learn a refinement mapping from the weak detector to category-aware classifiers, regressors, and segmentors $f_{\mathbf{W}^{*}}(\mathbf{z}); * \in \{cls, reg, seg\}$, which are used in the second stage of Faster / Mask R-CNN. Note that weak detectors simply output the proposal box of the pooled feature vector as the final location $\hat{\mathbf{y}} = \mathbf{rbox}$; while refined detectors are able to regress a better box. Here $f_{\mathbf{W}}(\cdot)$ is a learned neural network function parametrized by $\mathbf{W}$. We jointly train both branches and the RPN, and learning is divided into two stages: base-training and fine-tuning[3].

Base-training: During base-training, instances from $\mathcal{D}_{base}$ are used to obtain a detector / segmentation network for the *base* classes $\mathcal{C}_{base}$. Specifically, for each $b \in \mathcal{C}_{base}$, category-aware classifiers and regressors for the base classes are formulated as additive refinements to their corresponding weak counterparts. For region classifiers this takes the form of: $\hat{c} = \arg\max_{\mathcal{C}_{base}} \left[\texttt{softmax}\left(f_{\mathbf{W}^{cls}_{base}}(\mathbf{z})\right)\right]$, where

$$f_{\mathbf{W}^{cls}_{base}}(\mathbf{z}) = f_{\mathbf{W}^{weak}_{base}}(\mathbf{z}) + f_{\Delta\mathbf{W}^{cls}_{base}}(\mathbf{z}), \qquad (1)$$

where $f_{\Delta\mathbf{W}^{cls}_{base}}(\mathbf{z})$ is a zero-initialized residual to the logits of the weakly supervised detector. The regressed object location is similarly defined as:

$$\hat{\mathbf{y}} = \mathbf{rbox} + f_{\mathbf{W}^{reg}_{base}}(\mathbf{z}). \qquad (2)$$

Finally, as there is no estimate for the segmentation masks in the first stage of Mask R-CNN [20], $\hat{\mathbf{y}} = f_{\mathbf{W}^{seg}_{base}}(\mathbf{z})$ is a residual over $\mathbf{rbox}$ learned directly from *base* annotations.

Novel fine-tuning ($k > 0$): In the fine-tuning phase, the detectors / segmentors of the base classes are used to transfer information to the classes in $\mathcal{C}_{novel}$. The network is also fine-tuned on $\mathcal{D}_{novel}$, which, for a value of k, contains k bounding boxes / masks for novel and base classes. Here we consider the case of $k > 0$; we later address $k = 0$ case, which does not require fine-tuning. The key insight of our approach is to use additional *visual* and *lingual* similarities between the *novel* and *base* classes to enable effective transfer of the network onto the *novel* classes under varying degrees of supervision. Contrary to existing work [22, 56, 26] that only consider information from *base* category-aware classifiers, our approach additionally learns a mapping from *base* category-aware regressors and segmentors to obtain more accurate *novel* counterparts. For a specific proposal $\mathbf{rbox}$ with features $\mathbf{z}$, let $\mathbf{S}(\mathbf{z}) \in \mathbb{R}^{|\mathcal{C}_{novel}| \times |\mathcal{C}_{base}|}$ denote similarity between base classes and novel classes. The dependence on $\mathbf{z}$ stems from visual component of the similarity and is discussed in Section 4.2. Given this, for each proposal $\mathbf{z}$, the category-aware classifier for the novel classes is obtained as follows: $\hat{c} = \arg\max_{\mathcal{C}_{novel}} \left[\texttt{softmax}\left(f_{\mathbf{W}^{cls}_{novel}}(\mathbf{z})\right)\right]$, where $f_{\mathbf{W}^{cls}_{novel}}(\mathbf{z})$ can be written as,

$$\underbrace{f_{\mathbf{W}^{weak}_{novel}}(\mathbf{z})}_{\text{weak-detectors}} + \underbrace{\mathbf{S}(\mathbf{z})^{T} f_{\Delta\mathbf{W}^{cls}_{base}}(\mathbf{z})}_{\substack{\text{instance-level transfer}\\ \text{from base classes}}} + \underbrace{f_{\Delta\mathbf{W}^{cls}_{novel}}(\mathbf{z})}_{\substack{\text{instance-level}\\ \text{direct adaptation}}} \qquad (3)$$

where $\mathbf{S}(\mathbf{z}) = \texttt{softmax}(\mathbf{S}^{lin} \odot \mathbf{S}^{vis}(\mathbf{z}))$, which is computed along the columns in $\mathbf{S}(\mathbf{z})$, and $\odot$ denotes broadcast

[3] We use the nomenclature introduced in [24].

of vector similarity $\mathbf{S}^{vis}(\mathbf{z}) \in \mathbb{R}^{|\mathcal{C}_{base}|}$ followed by element-wise product with lingual similarity $\mathbf{S}^{lin} \in \mathbb{R}^{|\mathcal{C}_{novel}| \times |\mathcal{C}_{base}|}$. The interpretation of Eq.(3) is actually rather simple – we first refine the weak detectors for novel classes by similarity weighted additive refinements from base classes (*e.g.*, novel class `motorbike` may relay on base class `bicycle` for refinement; illustrations in supp. Sec. H.), denoted by "instance-level transfer from base classes". We then further directly adapt the resulting detector with few instances of the novel class (last term). Similarly, for each $\mathbf{z}$, the novel class object regressor can be obtained as,

$$\begin{aligned}\hat{\mathbf{y}} &= \mathbf{rbox} + f_{\mathbf{W}^{reg}_{novel}}(\mathbf{z}) \\ &= \mathbf{rbox} + \underbrace{\mathbf{S}^T(\mathbf{z}) f_{\mathbf{W}^{reg}_{base}}(\mathbf{z})}_{\substack{\text{instance-level transfer} \\ \text{from base classes}}} + \underbrace{f_{\Delta\mathbf{W}^{reg}_{novel}}(\mathbf{z})}_{\substack{\text{instance-level} \\ \text{direct adaptation}}}\end{aligned} \tag{4}$$

Finally, the segmentation head $f_{\mathbf{W}^{seg}_{novel}}(\mathbf{z})$ can be obtained as follows (additional details in appendix Section A),

$$\hat{\mathbf{y}} = f_{\mathbf{W}^{seg}_{novel}}(\mathbf{z}) = \underbrace{\mathbf{S}^T(\mathbf{z}) f_{\mathbf{W}^{seg}_{base}}(\mathbf{z})}_{\substack{\text{instance-level transfer} \\ \text{from base classes}}} + \underbrace{f_{\Delta\mathbf{W}^{seg}_{novel}}(\mathbf{z})}_{\substack{\text{instance-level} \\ \text{direct adaptation}}} \tag{5}$$

Semi-supervised zero-shot ($k = 0$): As we mentioned previously, our model is also readily applicable when $\mathcal{C}_{novel} = \emptyset$. This is a special case of the formulation above, where fine-tuning is not necessary or possible, and we only rely on base training and apply novel class evaluation procedure. The predictions for novel classes can be done as in Eq.(3), Eq.(4), and Eq.(5), but omitting the "instance-level direct adaptation" term in all three cases.

4.1. Weakly-Supervised Detector

As mentioned earlier, our approach leverages detectors trained on image level annotations to learn a mapping to supervised detectors/segmentors. We highlight that our approach is agnostic to the method used to train the weakly-supervised detector, and most of the existing approaches [2, 5, 54, 55] can be integrated into our framework. We, however, use the Online Instance Classifier Refinement (OICR) architecture proposed by Tang *et al.* [55] due to its simple architecture. OICR has R "refinement" modules $f_{\mathbf{W}^{weak}_r}(\mathbf{z})$ that progressively improve the detection quality. These individual "refinement" modules are combined to obtain the final prediction as follows,

$$\hat{\mathbf{a}} = \texttt{softmax}\left[f_{\mathbf{W}^{weak}}(\mathbf{z})\right] = \texttt{softmax}\left[\frac{1}{R}\sum_r f_{\mathbf{W}^{weak}_r}(\mathbf{z})\right] \tag{6}$$

We use the same loss formulation $\mathcal{L}^{weak}(\mathbf{a}, \hat{\mathbf{a}})$ described in [55], which compares predicted $(\hat{\mathbf{a}})$ and ground truth $(\mathbf{a})$ class labels, to train the OICR module (see Sect. 4.3). For additional details, we refer the reader to [55].

4.2. Similarity Matrices

As described in Eq.(3), (4), (5), the key contribution of our approach is the ability to semantically decompose the classifiers, detectors and segmentors of novel classes into their base classes' counterparts. To this end, we define a proposal-aware similarity $\mathbf{S}(\mathbf{z}) \in \mathbb{R}^{|\mathcal{C}_{novel}| \times |\mathcal{C}_{base}|}$, where each element captures the semantic similarity of novel class n to base class b. We assume $\mathbf{S}(\mathbf{z})$ can be decomposed into two components: *lingual* $\mathbf{S}^{lin}$ and *visual* $\mathbf{S}^{vis}(\mathbf{z})$ similarity.

Lingual Similarity: This term captures linguistic similarity between novel and base class labels. The intuition lies in the observation that semantically similar classes often have correlated occurrences in textual data. For a novel class n and a base class b, $\mathbf{S}^{lin}_{n,b} = \mathbf{g}_n^\top \mathbf{g}_b$; $\mathbf{g}_n$ and $\mathbf{g}_b$ are 300-dimensional GloVe [38] vector embeddings for n and b respectively[4].

Visual Similarity: Complementary to the lingual component, this *proposal-aware* similarity models the visual likeness of a proposal $\mathbf{z}$ to *base* class objects. For each $\mathbf{z}$, we use the normalized predictions $\hat{\mathbf{a}}$ of the weak detector $f_{\mathbf{W}^{weak}}(\mathbf{z})$ (Eq. (6)) as a proxy for the likelihood of $\mathbf{z}$ belonging to a *base* class b. Specifically, let $\hat{\mathbf{a}}_b$ be the score corresponding to the base class b. For a novel class n and a base class b, the visual similarity $\mathbf{S}^{vis}_{n,b}(\mathbf{z})$ is then defined as,

$$\mathbf{S}^{vis}_{n,b}(\mathbf{z}) = \frac{\hat{\mathbf{a}}_b}{\sum_{i \in base} \hat{\mathbf{a}}_i} \tag{7}$$

Note, computing this visual similarity *does not* require learning additional parameters. Rather, it is just a convenient by-product of training our model. As a result, this similarity can be efficiently computed. Our visual similarity formulation, in its essence, is similar to the one used in [56]. However, [56] use *image-level* scores aggregated over a validation set, lacking ability to adapt to a specific proposal. Additionally, our framework is extremely flexible and can easily utilize any additional information, akin to [26], to obtain a more accurate semantic decomposition $\mathbf{S}(\mathbf{z})$. However, as computing these might require additional datasets and pre-trained models, we refrain from incorporating them into our model.

4.3. Training

We now describe the optimization objective used to train our proposed approach in an end-to-end fashion. During base training, the objective can be written as,

$$\mathcal{L}^t = \mathcal{L}^{rcnn} + \alpha \mathcal{L}^{weak} \tag{8}$$

where $\mathcal{L}^{rcnn}$ is the Faster/Mask R-CNN [20, 43] objective, and $\mathcal{L}^{weak}$ is the OICR [55] objective; $\alpha = 1$ is the weighting hyperparameter. In fine-tuning, we refine the model only using $\mathcal{L}^{rcnn}$. Note, our approach affords the flexibility to either use pretrained proposals or jointly train a RPN during

[4] For class names that contain multiple words, we average individual GloVe word embeddings.

the *base-training* phase using instance-level *base* class annotations. Fine-tuning only effects last term of Eq.(3), (4), and (5), while everything else is optimized using base training objective. Further details are in suppl. Sec. B.

5. Experiments

We evaluate our approach against related methods in the semi-supervised and few-shot domain. Comparison against work in the weakly-supervised literature is provided in supplementary Sec. E. Note, for *base* classes, across all experiments, the *same* images are used for both image and instance level annotations. This does not induce any additional cost as instance-level labels implicitly give image-level labels.

5.1. Semi-supervised Object Detection

Datasets. We evaluate the performance of our framework on MSCOCO [32] 2015 and 2017 datasets. Similar to [17, 26], we divide the 80 object categories into 20 *base* and 60 *novel* classes, where the *base* classes are identical to the 20 VOC [14] categories. For our model and the baselines, we assume image-level supervision for *all* 80 classes, whereas instance-level supervision is only available for 20 *base* classes. For few-shot experiments ($k > 0$) we additionally assume k instance-level annotations for the *novel* classes.

Semi-supervised zero-shot ($k = 0$). Table 1 compares the performance of our proposed approach against the most relevant semi-supervised zero-shot ($k = 0$) methods [22, 23, 26, 56] on *novel* classes. As an upper-bound, we also show the performance of a fully-supervised model. To ensure fair comparison, we follow the experimental setting in the strongest baseline DOCK [26], and borrow performance for [22, 23, 56] from their paper. All models are trained using the *same* backbone: VGG-CNN-F [6] which is pretrained on the ImageNet dataset [11]. Similar to [26], we use the MCG [39] proposals instead of training the RPN. The models are evaluated using mAP at IoU threshold 0.5 denoted as AP_{50}.

UniT beats the closest baseline, DOCK [26], by a significant margin ($\sim$16% on AP_{50}), despite DOCK using more sophisticated similarity measures for knowledge transfer, which require additional data from VOC [14], Visual Genome [25], and SUN [66] datasets. As DOCK only transfers knowledge from *base* class classifiers, this performance gap can be attributed to UniT additionally effectively transferring knowledge from *base* class regressors onto *novel* class regressors (Eq. 4). Note, our work is complimentary to DOCK. Their richer similarity measures can be easily integrated into our framework by modifying $\mathbf{S}(\mathbf{z})$ (Sec. 4.2).

Semi-supervised few-shot ($k > 0$). Table 2 compares the performance of our method with NOTE-RCNN [17], which is the only relevant baseline under this setting, on *novel* classes. We follow the experimental setting described in [17], and our model is trained using the same backbone as NOTE-RCNN: Inception-Resnet-V2 [53] pretrained on the ImageNet classification dataset [11], where the RPN is learned from the instance-level *base* data. Similar to [17], we assume k instance-level annotations for the *novel* classes, where $k \in \{12, 33, 55, 76, 96\}$. To ensure fair comparison, the performance of NOTE-RCNN [17] is taken from their published work[5]. We report mAP on novel classes averaged over IoU thresholds in $[0.5 : 0.05 : 0.95]$.

UniT outperforms NOTE-RCNN [17] on all values of k, providing an improvement of up to $\sim$23%. Contrary to NOTE-RCNN that only trains *novel* regressors on the k shots, UniT benefits from effectively mapping information from *base* regressors to *novel* regressors. In addition, UniT also has the advantage of allowing end-to-end training while simultaneously being simple and interpretable. NOTE-RCNN, on the other hand, employs a complex multi-step bounding box mining framework that takes longer to train on *novel* classes. Note that, in principle, one could incorporate the box mining mechanism into our framework as well.

Method	AP_{50}	AP_S	AP_M	AP_L
LSDA [22]	4.6	1.2	5.1	7.8
LSDA+Semantic [56]	4.7	1.1	5.1	8.0
LSDA+MIL [23]	5.9	1.5	8.3	10.7
DOCK [26]	14.4	2.0	12.8	24.9
UniT (Ours)	**16.7**	**3.2**	**16.6**	**27.3**
Full Supervision [26]	25.2	5.8	26.0	41.6

Table 1: **Comparison to semi-supervised zero-shot.** All models are trained on VGG-CNN-F [6] backbone.

Method / Shots (k)	12	33	55	76	96
NOTE-RCNN [17]	14.1	14.2	17.1	19.8	19.9
UniT (Ours)	**14.7**	**17.4**	**19.3**	**20.9**	**22.1**

Table 2: **Comparison to semi-supervised few-shot.** All models are trained on Inception-ResNet-v2 [53] backbone. Mean Average Precision (mAP) on novel classes averaged over IoU thresholds in $[0.5 : 0.05 : 0.95]$ is reported.

5.2. Few-shot Object Detection and Segmentation

Datasets. We evaluate our models on VOC 2007 [14], VOC 2012 [13], and MSCOCO [32], as used in previous few-shot object detection and segmentation works [24, 62, 63, 67]. For both detection and segmentation, we consistently follow the data splits introduced and used in [24, 67]. In case of VOC, we use VOC 07 test set (5k images) for evaluation and VOC 07+12 trainval sets (16.5k images) for training. The 20 object classes are divided into 3 different class split sets, each with 15 base and 5 novel classes. For novel classes, images provided by Kang *et al.* [24] are used for k-shot fine-tuning.We report mean Average Precision (mAP) on novel classes and use a standard IoU threshold of 0.5 [14]. For MSCOCO [32], consistent with [24], we use 5k images from the validation set for evaluation and the remaining 115k trainval images for training. We assign 20 object classes from VOC as the novel classes and remaining 60 as the base

[5] [17] report results as a plot instead of listing the raw values. As the authors were unreachable, Table 2 lists our best interpretation of the plot.

		Novel Set 1						Novel Set 2						Novel Set 3					
Method / Shots		0	1	2	3	5	10	0	1	2	3	5	10	0	1	2	3	5	10
Joint	FRCN [67]	-	2.7	3.1	4.3	11.8	29.0	-	1.9	2.6	8.1	9.9	12.6	-	5.2	7.5	6.4	6.4	6.4
Transfer	FRCN [62]	-	15.2	20.3	29.0	40.1	45.5	-	13.4	20.6	28.6	32.4	38.8	-	19.6	20.8	28.7	42.2	42.1
Few-Shot	Kang *et al.* [24]	-	14.8	15.5	26.7	33.9	47.2	-	15.7	15.3	22.7	30.1	39.2	-	19.2	21.7	25.7	40.6	41.3
	Wang *et al.* [63]	-	18.9	20.6	30.2	36.8	49.6	-	21.8	23.1	27.8	31.7	43.0	-	20.6	23.9	29.4	43.9	44.1
	Yan *et al.* [67]	-	19.9	25.5	35.0	45.7	51.5	-	10.4	19.4	29.6	34.8	45.4	-	14.3	18.2	27.5	41.2	48.1
	Wang *et al.* [62]	-	39.8	36.1	44.7	55.7	56.0	-	23.5	26.9	34.1	35.1	39.1	-	30.8	34.8	42.8	49.5	49.8
Semi+Any Shot	UniT (Ours)	**75.6**	**75.7**	**75.8**	**75.9**	**76.1**	**76.7**	**56.9**	**57.2**	**57.4**	**57.9**	**58.2**	**63.0**	**67.5**	**67.6**	**68.1**	**68.2**	**68.6**	**70.0**
Fully-supervised	FRCN	84.71						82.89						82.57					

Table 3: **Few-shot object detection on VOC.** FRCN = Faster R-CNN with ResNet-101 backbone. Mean AP_{50} reported on *novel* classes; performance on *base* classes is reported in supplementary Section I.

#Shots		AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
$k = 0$	UniT (Ours)	18.9	36.1	17.5	8.7	20.4	27.6
$k = 10$	Transfer: FRCN [67]	6.5	13.4	5.9	1.8	5.3	11.3
	Kang *et al.* [24]	5.6	12.3	4.6	0.9	3.5	10.5
	Wang *et al.* [63]	7.1	14.6	6.1	1.0	4.1	12.2
	Yan *et al.* [67]	8.7	19.1	6.6	2.3	7.7	14.0
	Wang *et al.* [62]	10.0	-	9.3	-	-	-
	UniT (Ours)	**21.7**	**40.8**	**20.6**	**9.1**	**23.8**	**31.3**

Table 4: **Few-shot object detection on COCO**. FRCN using ResNet-50 backbone. Full table in suppl. Section C.

#Shots	Method		AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
$k = 0$	UniT (Ours)	Box	20.2	36.8	19.5	8.5	20.9	28.9
		Mask	17.6	32.7	17.0	5.6	17.6	27.7
$k = 10$	Yan *et al.* [67]	Box	5.6	14.2	3.0	2.0	6.6	8.8
		Mask	4.4	10.6	3.3	0.5	3.6	7.2
	UniT (Ours)	Box	**22.8**	**41.6**	**21.9**	**9.4**	**24.4**	**32.3**
		Mask	**20.5**	**38.6**	**19.7**	**6.0**	**20.5**	**31.8**

Table 5: **Few-shot instance segmentation on COCO**. Complete table is in supplementary Section D.

classes. We report the standard evaluation metric on COCO [43]. In line with the baselines, for both VOC and MSCOCO, the RPN is trained jointly using *base* class annotations.

PASCAL VOC Detection. Table 3 summarizes the results on VOC for three different novel class splits with different k-shot settings. Following [62, 67], UniT assumes Faster R-CNN [43] with an ImageNet [45] pretrained ResNet-101 [21] backbone. UniT outperforms the related state-of-the-art methods on all values of k, including the scenario with no *novel* class instance-level supervision ($k = 0$), showing the effectiveness of transfer from base to novel classes. As UniT uses additional weak image-level data for *novel* classes, this is not an equivalent comparison (see Sec. 5.3 for comparisons under similar annotation budget). However, we highlight that such data is readily available, cheaper to obtain [4], and provides significant performance improvements.

MS-COCO Detection. Table 4 describes the results on COCO dataset. Similar to [67, 62], we use ImageNet [11] pretrained ResNet-50 [21] as the backbone. We observe similar trends as above. In addition, our performance consistently increases with the value of k even on larger datasets, showing that UniT is effective and can easily scale to different amounts of instance-level supervision. The full table is in suppl. Sec. C. Figure 3 shows qualitative results, indicating our method is able to correctly detect *novel* classes.

MS-COCO Segmentation. Table 5 summarizes the results. Similar to [67], we choose an ImageNet[11] pretrained ResNet-50 [21] backbone. UniT consistently improves over [67], demonstrating that our approach is not limited to bounding boxes, and is able to generalize over the type of downstream structured label by effectively transferring information from *base* segmentations to *novel* segmentations. The full table is provided in supplementary Section D. Figure 3 shows some qualitative results on $k = 0$ for *novel* classes.

Ablation. A complete ablation study on MSCOCO [32] is provided in supplementary Section G. We report performance on the novel split used by [67], starting with only weak detectors and progressively adding the terms in Eq.(1), (3), (4), and (5). Weighting with visual and lingual similarity results in $+1.4$ AP_{50} improvement (Eq. (3)), transfer from *base* regressors (Eq. (4)) provides an additional $+7$ AP_{50} imrovement. Finally, transfer from *base* class segmentations (Eq. (5)) leads to an added gain of $+7.5$ on mask AP_{50}.

5.3. Limited Annotation Budget

Compared to approaches in the few-shot detection (and segmentation) domain like [24, 62, 63, 67], UniT assumes additional image-level annotations for *novel* classes. We argue this is a reasonable assumption considering that such annotations are readily available in abundance for thousands of object classes ($\sim$22K in ImageNet [11] and $\sim$20K in Open Image v4 dataset [28]). Experiments in Section 5.2 further highlight the performance improvements possible by using such inexpensive data. However, this raises an interesting question as to what form of supervision is more valuable, if one is to collect it. To experiment with this, we conceptually impose an annotation budget that limits the number of *novel* class image-level annotations our approach can use. For object detection on VOC [13], we assume 7 image-level annotations can be generated in the same time as 1 instance-level annotation. This conversion factor of 7 is

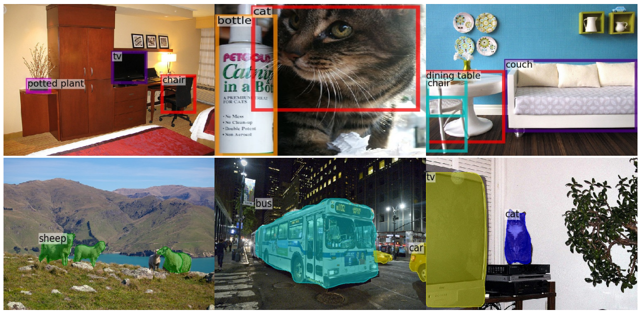

Figure 3: **Qualitative Visualizations.** Semi-supervised zero-shot ($k = 0$) detection (top) and instance segmentation (bottom) performance on *novel* classes in MS-COCO (color = category). See suppl. Section J for more examples.

motivated by the timings reported in [4] and is a *conservative* estimate (details in suppl. Sec. F)[6]. For each value of k in a few-shot setup, we train a variant of UniT, referred to as UniT$_{budget=k}$, that only assumes $7 \times k$ image-level annotations for *novel* classes. We then compare the *zero-shot* performance of UniT$_{budget=k}$ against the corresponding k-shot object detection benchmarks[7] reported in [62]. Note, UniT$_{budget=k}$ assumes abundant image-level annotations for *base* classes. However, as the same images are used for both instance and image level annotations, this does not impose any additional annotation cost when compared to baselines. This setting enables apples-to-apples comparisons with the baselines, while simultaneously contrasting the relative importance of image-level and instance-level annotations.

Please refer to Section 5.2 for details on the dataset and setup. Table 6 summarizes the results on VOC for three novel class splits assuming different k-shot settings. Following [62], all models use ResNet-101 [21] as the backbone. For each split and k-shot, 10 repeated runs of UniT$_{budget=k}$ are averaged, each trained by selecting a different set of $7 \times k$ weakly-labelled *novel* class images. For a fixed budget, equivalent to 10 instance-level annotations, we further analyze the relative importance of the two types of annotations by varying the proportions of image and instance-level annotations used. This is summarized in Table 7 for the first *novel* split. Even under equal budget constraints, UniT$_{budget=k}$ outperforms the state-of-the-art [62] on multiple splits. This highlights three key observations: i) image-level supervision, which is cheaper to obtain [4], provides a greater 'bang-for-the-buck' compared to instance-level supervision, ii) our structured transfer from *base* classes is effective even under limited *novel* class supervision, and iii) from Table 7, in a low-shot and fixed budget setting, it is more beneficial to just use weak supervision, instead of some combination of both. Furthermore, as our approach is agnostic to the type of weak detector used, employing better weak detectors like [54, 2] could further improve the performance of UniT$_{budget=k}$.

#Shots	Method	Split 1	Split 2	Split 3
1	Kang *et al.* [24]	14.2 ± 1.7	12.3 ± 1.9	12.5 ± 1.6
	Wang *et al.* [62]	25.3 ± 2.2	$\mathbf{18.3 \pm 2.4}$	17.9 ± 2.0
	UniT$_{budget=1}$ (Ours)	$\mathbf{28.3 \pm 2.0}$	17.0 ± 1.9	$\mathbf{26.2 \pm 2.5}$
5	Kang *et al.* [24]	36.5 ± 1.4	31.4 ± 1.5	33.8 ± 1.4
	Wang *et al.* [62]	47.9 ± 1.2	34.1 ± 1.4	40.8 ± 1.4
	UniT$_{budget=5}$ (Ours)	$\mathbf{50.9 \pm 1.4}$	$\mathbf{36.2 \pm 1.7}$	$\mathbf{47.4 \pm 1.2}$
10	Wang *et al.* [62]	52.8 ± 1.0	39.5 ± 1.1	45.6 ± 1.1
	UniT$_{budget=10}$ (Ours)	$\mathbf{59.0 \pm 1.5}$	$\mathbf{40.8 \pm 1.3}$	$\mathbf{52.9 \pm 1.1}$

Table 6: **Limited annotation budget.** Averaged AP$_{50}$ for 10 random runs with 95% confidence interval estimate [62].

Method	Weak Anno.(%)	Instance Anno.(%)	AP$_{50}$
Wang *et al.* [62] + 10-Shots	0	100	52.8 ± 1.0
UniT$_{budget=1}$ + 9-Shots	10	90	49.2 ± 0.6
UniT$_{budget=5}$ + 5-Shots	50	50	54.0 ± 0.8
UniT$_{budget=10}$ + 0-Shots	100	0	$\mathbf{59.0 \pm 1.5}$

Table 7: **Using different annotation proportions**. For the same budget, we vary the amount of image/instance level annotation. Averaged AP$_{50}$ for 10 random runs with 95% confidence interval estimate of the mean values [62] is shown.

6. Discussion and Conclusion

We propose an intuitive semi-supervised model that is applicable to a wide range of supervision: from zero to a few instance-level samples per *novel* class. For *base* classes, our model learns a mapping from weakly-supervised to fully-supervised detectors/segmentors. By leveraging similarities between the *novel* and *base* classes, we transfer those mappings to obtain detectors/segmentors for *novel* classes; refining them with a few novel class instance-level annotated samples, if available. This versatile paradigm works significantly better than traditional semi-supervised and few-shot detection and segmentation methods.

Acknowledgments and Disclosure of Funding

This work was funded, in part, by the Vector Institute for AI, Canada CIFAR AI Chair, NSERC CRC and an NSERC DG and Accelerator Grants. This material is based upon work supported by the US Air Force Research Laboratory (AFRL) under the DARPA Learning with Less Labels (LwLL) program (Contract No.FA8750-19-C-0515) [8].

Hardware resources used in preparing this research were provided, in part, by the Province of Ontario, the Government of Canada through CIFAR, and companies sponsoring the Vector Institute[9]. Additional support was provided by JELF CFI grant and Compute Canada under the RAC award.

Finally, we sincerely thank Bicheng Xu, Issam Laradji and Prof. Frank Wood for valuable feedback and discussions.

[6] This factor is expected to be higher in practice, as we don't consider situations where boxes/masks are rejected and need to be redrawn [36].

[7] These benchmarks use multiple random splits as opposed to curated splits used in [24] and Table 3. As per [62], this helps reduce variance.

[8] The views and conclusions contained herein are those of the authors and should not be interpreted as representing the official policies or endorsements, either expressed or implied, of DARPA or the U.S. Government.

[9] www.vectorinstitute.ai/#partners

References

[1] Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando De Freitas. Learning to learn by gradient descent by gradient descent. In *Advances in neural information processing systems*, pages 3981–3989, 2016.

[2] Aditya Arun, CV Jawahar, and M Pawan Kumar. Dissimilarity coefficient based weakly supervised object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9432–9441, 2019.

[3] Ankan Bansal, Karan Sikka, Gaurav Sharma, Rama Chellappa, and Ajay Divakaran. Zero-shot object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 384–400, 2018.

[4] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei. Whats the point: Semantic segmentation with point supervision. In *ECCV*, 2016.

[5] Hakan Bilen and Andrea Vedaldi. Weakly supervised deep detection networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2846–2854, 2016.

[6] Ken Chatfield, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Return of the devil in the details: Delving deep into convolutional nets. *BMVC*, 2014.

[7] Hao Chen, Yali Wang, Guoyou Wang, and Yu Qiao. Lstd: A low-shot transfer detector for object detection. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[8] Liang-Chieh Chen, Alexander Hermans, George Papandreou, Florian Schroff, Peng Wang, and Hartwig Adam. Masklab: Instance segmentation by refining object detection with semantic and direction features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4013–4022, 2018.

[9] Ramazan Gokberk Cinbis, Jakob Verbeek, and Cordelia Schmid. Weakly supervised object localization with multi-fold multiple instance learning. *IEEE transactions on pattern analysis and machine intelligence*, 39(1):189–203, 2016.

[10] Berkan Demirel, Ramazan Gokberk Cinbis, and Nazli Ikizler-Cinbis. Zero-shot object detection by hybrid region embedding. *arXiv preprint arXiv:1805.06157*, 2018.

[11] 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.

[12] Ali Diba, Vivek Sharma, Ali Pazandeh, Hamed Pirsiavash, and Luc Van Gool. Weakly supervised cascaded convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 914–922, 2017.

[13] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111(1):98–136, 2015.

[14] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[15] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1126–1135. JMLR. org, 2017.

[16] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In *Advances in neural information processing systems*, pages 2121–2129, 2013.

[17] Jiyang Gao, Jiang Wang, Shengyang Dai, Li-Jia Li, and Ram Nevatia. Note-rcnn: Noise tolerant ensemble rcnn for semi-supervised object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 9508–9517, 2019.

[18] Yan Gao, Boxiao Liu, Nan Guo, Xiaochun Ye, Fang Wan, Haihang You, and Dongrui Fan. C-midn: Coupled multiple instance detection network with segmentation guidance for weakly supervised object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9834–9843, 2019.

[19] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[20] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[22] Judy Hoffman, Sergio Guadarrama, Eric S Tzeng, Ronghang Hu, Jeff Donahue, Ross Girshick, Trevor Darrell, and Kate Saenko. Lsda: Large scale detection through adaptation. In *Advances in Neural Information Processing Systems*, pages 3536–3544, 2014.

[23] Judy Hoffman, Deepak Pathak, Trevor Darrell, and Kate Saenko. Detector discovery in the wild: Joint multiple instance and representation learning. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 2883–2891, 2015.

[24] Bingyi Kang, Zhuang Liu, Xin Wang, Fisher Yu, Jiashi Feng, and Trevor Darrell. Few-shot object detection via feature reweighting. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[25] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73, 2017.

[26] Krishna Kumar Singh, Santosh Divvala, Ali Farhadi, and Yong Jae Lee. Dock: Detecting objects by transferring common-sense knowledge. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 492–508, 2018.

[27] Alina Kuznetsova, Sung Ju Hwang, Bodo Rosenhahn, and Leonid Sigal. Expanding object detector's horizon: Incremental learning framework for object detection in videos. In

Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 28–36, 2015.

[28] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4. *International Journal of Computer Vision*, pages 1–26, 2020.

[29] Issam H Laradji, David Vazquez, and Mark Schmidt. Where are the masks: Instance segmentation with image-level supervision. In *BMVC*, 2019.

[30] Zhihui Li, Lina Yao, Xiaoqin Zhang, Xianzhi Wang, Salil Kanhere, and Huaxiang Zhang. Zero-shot object detection with textual descriptions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8690–8697, 2019.

[31] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[32] 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 *European conference on computer vision*, pages 740–755. Springer, 2014.

[33] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[34] Tianxiang Pan, Bin Wang, Guiguang Ding, Jungong Han, and Jun-Hai Yong. Low shot box correction for weakly supervised object detection. In *IJCAI*, pages 890–896, 2019.

[35] Dim P Papadopoulos, Alasdair DF Clarke, Frank Keller, and Vittorio Ferrari. Training object class detectors from eye tracking data. In *European conference on computer vision*, pages 361–376. Springer, 2014.

[36] Dim P Papadopoulos, Jasper RR Uijlings, Frank Keller, and Vittorio Ferrari. Training object class detectors with click supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6374–6383, 2017.

[37] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8024–8035, 2019.

[38] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[39] Jordi Pont-Tuset, Pablo Arbelaez, Jonathan T Barron, Ferran Marques, and Jitendra Malik. Multiscale combinatorial grouping for image segmentation and object proposal generation. *IEEE transactions on pattern analysis and machine intelligence*, 39(1):128–140, 2016.

[40] Shafin Rahman, Salman Khan, and Fatih Porikli. Zero-shot object detection: Learning to simultaneously recognize and localize novel concepts. In *Asian Conference on Computer Vision*, pages 547–563. Springer, 2018.

[41] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *ICLR*, 2017.

[42] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.

[43] 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*, pages 91–99, 2015.

[44] Zhongzheng Ren, Zhiding Yu, Xiaodong Yang, Ming-Yu Liu, Yong Jae Lee, Alexander G Schwing, and Jan Kautz. Instance-aware, context-focused, and memory-efficient weakly supervised object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10598–10607, 2020.

[45] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[46] Arka Sadhu, Kan Chen, and Ram Nevatia. Zero-shot grounding of objects from natural language queries. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4694–4703, 2019.

[47] Jürgen Schmidhuber. *Evolutionary principles in self-referential learning, or on learning how to learn: the meta-meta-... hook*. PhD thesis, Technische Universität München, 1987.

[48] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[49] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in neural information processing systems*, pages 4077–4087, 2017.

[50] Hyun Oh Song, Yong Jae Lee, Stefanie Jegelka, and Trevor Darrell. Weakly-supervised discovery of visual pattern configurations. In *Advances in Neural Information Processing Systems*, pages 1637–1645, 2014.

[51] Hao Su, Jia Deng, and Li Fei-Fei. Crowdsourcing annotations for visual object detection. In *Workshops at the Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.

[52] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE international conference on computer vision*, pages 843–852, 2017.

[53] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alex Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. *AAAI*, 2017.

[54] Peng Tang, Xinggang Wang, Song Bai, Wei Shen, Xiang Bai, Wenyu Liu, and Alan Yuille. Pcl: Proposal cluster learning for weakly supervised object detection. *IEEE transactions on pattern analysis and machine intelligence*, 42(1):176–191, 2018.

[55] Peng Tang, Xinggang Wang, Xiang Bai, and Wenyu Liu. Multiple instance detection network with online instance classifier refinement. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2843–2851, 2017.

[56] Yuxing Tang, Josiah Wang, Boyang Gao, Emmanuel Dellandréa, Robert Gaizauskas, and Liming Chen. Large scale semi-supervised object detection using visual and semantic knowledge transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2119–2128, 2016.

[57] Sebastian Thrun and Lorien Pratt. *Learning to learn*. Springer Science & Business Media, 2012.

[58] Jasper Uijlings, Stefan Popov, and Vittorio Ferrari. Revisiting knowledge transfer for training object class detectors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1101–1110, 2018.

[59] Jasper RR Uijlings, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *International journal of computer vision*, 104(2):154–171, 2013.

[60] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *Advances in neural information processing systems*, pages 3630–3638, 2016.

[61] Jiajie Wang, Jiangchao Yao, Ya Zhang, and Rui Zhang. Collaborative learning for weakly supervised object detection. *arXiv preprint arXiv:1802.03531*, 2018.

[62] Xin Wang, Thomas E Huang, Trevor Darrell, Joseph E Gonzalez, and Fisher Yu. Frustratingly simple few-shot object detection. *ICML*, 2020.

[63] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Meta-learning to detect rare objects. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[64] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. `https://github.com/facebookresearch/detectron2`, 2019.

[65] Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly. *IEEE transactions on pattern analysis and machine intelligence*, 41(9):2251–2265, 2018.

[66] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010.

[67] Xiaopeng Yan, Ziliang Chen, Anni Xu, Xiaoxi Wang, Xiaodan Liang, and Liang Lin. Meta r-cnn: Towards general solver for instance-level low-shot learning. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[68] Hao Yang, Hao Wu, and Hao Chen. Detecting 11k classes: Large scale object detection without fine-grained bounding boxes. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9805–9813, 2019.

[69] Eloi Zablocki, Patrick Bordes, Benjamin Piwowarski, Laure Soulier, and Patrick Gallinari. Context-aware zero-shot learning for object recognition. *arXiv preprint arXiv:1904.12638*, 2019.

[70] Zhaoyang Zeng, Bei Liu, Jianlong Fu, Hongyang Chao, and Lei Zhang. Wsod2: Learning bottom-up and top-down objectness distillation for weakly-supervised object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8292–8300, 2019.

[71] Yanzhao Zhou, Yi Zhu, Qixiang Ye, Qiang Qiu, and Jianbin Jiao. Weakly supervised instance segmentation using class peak response. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3791–3800, 2018.

[72] Pengkai Zhu, Hanxiao Wang, and Venkatesh Saligrama. Dont even look once: Synthesizing features for zero-shot detection. *arXiv preprint arXiv:1911.07933*, 2019.

[73] C Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *European conference on computer vision*, pages 391–405. Springer, 2014.