Context-Transformer: Tackling Object Confusion for Few-Shot Detection

Ze Yang,[1*] Yali Wang,[1*] Xianyu Chen,[1] Jianzhuang Liu,[2] Yu Qiao[1,3†]
[1]ShenZhen Key Lab of Computer Vision and Pattern Recognition, SIAT-SenseTime Joint Lab,
Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences
[2]Huawei Noah's Ark Lab
[3]SIAT Branch, Shenzhen Institute of Artificial Intelligence and Robotics for Society
{ze.yang, yali.wang, yu.qiao}@siat.ac.cn, xianyuchen1992@outlook.com, liu.jianzhuang@huawei.com

Abstract

Few-shot object detection is a challenging but realistic scenario, where only a few annotated training images are available for training detectors. A popular approach to handle this problem is transfer learning, i.e., fine-tuning a detector pretrained on a source-domain benchmark. However, such transferred detector often fails to recognize new objects in the target domain, due to low data diversity of training samples. To tackle this problem, we propose a novel Context-Transformer within a concise deep transfer framework. Specifically, Context-Transformer can effectively leverage source-domain object knowledge as guidance, and automatically exploit contexts from only a few training images in the target domain. Subsequently, it can adaptively integrate these relational clues to enhance the discriminative power of detector, in order to reduce object confusion in few-shot scenarios. Moreover, Context-Transformer is flexibly embedded in the popular SSD-style detectors, which makes it a plug-and-play module for end-to-end few-shot learning. Finally, we evaluate Context-Transformer on the challenging settings of few-shot detection and incremental few-shot detection. The experimental results show that, our framework outperforms the recent state-of-the-art approaches.

1 Introduction

Object detection has been mainly promoted by deep learning frameworks (Ren et al. 2015; He et al. 2017; Redmon et al. 2016; Liu et al. 2016). However, the impressive performance of these detectors heavily relies on large-scale benchmarks with bounding box annotations, which are time-consuming or infeasible to obtain in practice. As a result, we often face a real-world scenario, i.e., few-shot object detection, where there are only a few annotated training images. In this case, deep learning will deteriorate due to severe overfitting.

A popular strategy is transfer learning, i.e., one can train an object detector with a large-scale benchmark in the source domain, and then fine-tune it with a few samples in the target domain. By doing so, we observe an interesting and important phenomenon. For few-shot object detection, a transferred detector often performs well on localization while encounters difficulty in classification, e.g., a *horse* is well-localized but misclassified as a *dog* in Fig. 1.

The main reason is that, an object detector uses bounding box regressor (BBOX) for localization while object+background classifier (OBJ+BG) for classification. BBOX is often category-irrelevant. Hence, we can use source-domain BBOX as a reliable initialization of target-domain BBOX. In this case, the detector can effectively localize new objects after fine-tuning with a few training samples in the target domain. On the contrary, OBJ+BG is category-specific. In other words, it has to be randomly initialized for new categories in the target domain. However, only a few training images are available in this domain. Such low data diversity significantly enlarges the training difficulty of classifier, which leads to the key problem above, i.e., object confusion caused by annotation scarcity.

To address this problem, we propose a novel Context-Transformer. It can automatically exploit contexts from only a few images on hand, and attentively integrate such distinct clues to generalize detection. Our design is inspired by (Oliva and Torralba 2007) that, at an early age with little object knowledge, humans can build the contextual associations for visual recognition. In other words, under little supervision scenarios, we will try to explore distinct clues in the surroundings (which we refer to as contextual fields in this paper), to clarify object confusion. For example, a few images may be discriminative enough to distinguish *horse* from *dog*, when we find that these images contain important contents such as a person sits on this animal, the scene is about wild grassland, etc.

To mimic this capacity, we design Context-Transformer in a concise transfer framework. Specifically, it consists of two simple but effective submodules, i.e., affinity discovery and context aggregation. For a target-domain image, affinity discovery first constructs a set of contextual fields, according to default prior boxes (also called as anchor boxes) in the detector. Then, it adaptively exploits relations between prior boxes and contextual fields. Finally, context aggregation leverages such relations as guidance, and integrates key contexts attentively into each prior box. As a

[*]Ze Yang and Yali Wang contribute equally
[†]Corresponding author.
Copyright © 2020, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

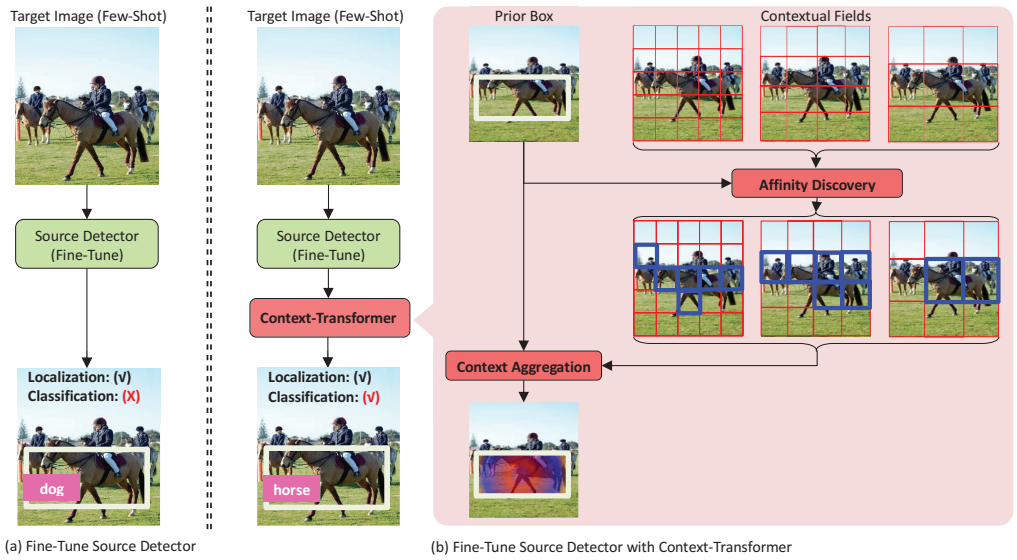

Figure 1: Our Motivation. Fine-tuning a pretrained detector is a popular approach for few-shot object detection. However, such transferred detector often suffers from object confusion in the new target domain, e.g., a *horse* is misclassified as a *dog*, due to annotation scarcity. Alternatively, humans can effectively correct such few-shot confusions by further exploiting discriminative context clues from only a few images on hand. Inspired by this observation, we introduce a novel Context-Transformer to tackle object confusion for few-shot object detection. More explanations can be found in Section 1 and 4.

result, Context-Transformer can generate a context-aware representation for each prior box, which allows detector to distinguish few-shot confusion with discriminative context clues. **To our best knowledge, Context-Transformer is the first work to investigate context for few-shot object detection**. Since it does not require excessive contextual assumptions on aspect ratios, locations and spatial scales, Context-Transformer can flexibly capture diversified and discriminative contexts to distinguish object confusion. More importantly, it leverages elaborative transfer insights for few-shot detection. With guidance of source-domain knowledge, Context-Transformer can effectively reduce learning difficulty when exploiting contexts from few annotated images in the target domain. Additionally, we embed Context-Transformer into the popular SSD-style detectors. Such plug-and-play property makes it practical for few-shot detection. Finally, we conduct extensive experiments on different few-shot settings, where our framework outperforms the recent state-of-the-art approaches.

2 Related Works

Over the past years, we have witnessed the fast development of deep learning in object detection. In general, the deep detection frameworks are mainly categorized into two types, i.e., one-stage detectors (e.g., YOLO or SSD styles (Redmon et al. 2016; Liu et al. 2016)) and two-stage detectors (e.g., R-CNN styles (Girshick et al. 2014; Girshick 2015; Ren et al. 2015; He et al. 2017)). Even though both types have achieved great successes in object detection, they heavily depend on large-scale benchmarks with bounding boxes annotations. Collecting such fully-annotated datasets is often difficult or labor-intensive for real-life applications.

Few-Shot Object Detection. To alleviate this problem, weakly (Bilen and Vedaldi 2016; Lai and Gong 2017; Tang et al. 2017) or semi (Hoffman et al. 2014; Tang et al. 2016) supervised detectors have been proposed. However, only object labels are available in the weakly-supervised setting, which restricts the detection performance. The semi-supervised detectors often assume that, there is a moderate amount of object box annotations, which can be still challenging to obtain in practice. Subsequently, few-shot data assumption has been proposed in (Dong et al. 2018). However, it relies on multi-model fusion with a complex training procedure, which may reduce the efficiency of model deployment for a new few-shot detection task. Recently, a feature reweighting approach has been introduced in a transfer learning framework (Kang et al. 2019). Even though its simplicity is attractive, this approach requires object masks as extra inputs to train a meta-model of feature reweighting. More importantly, most approaches may ignore object confusion caused by low data diversity. Alternatively, we propose a novel Context-Transformer to address this problem.

Object Detection with Contexts. Modeling context has been a long-term challenge for object detection (Bell et al. 2016; Chen and Gupta 2017; Kantorov et al. 2016; Mottaghi et al. 2014). The main reason is that, objects may have various locations, scales, aspect ratios, and classes. It is often difficult to model such complex instance-level relations by manual design. Recently, several works have been proposed to alleviate this difficulty, by automatically building up object relations with non-local attention (Wang et al. 2018; Hu et al. 2018). However, these approaches would lead to unsatisfactory performance in few-shot detection, without elaborative transfer insights. Alternatively, our

Context-Transformer is built upon a concise transfer framework, which can leverage source-domain object knowledge as guidance, and effectively exploit target-domain context for few-shot generalization.

Few-Shot Learning. Unlike deep learning models, humans can learn new concepts with little supervision (Lake, Salakhutdinov, and Tenenbaum 2015). For this reason, few-shot learning has been investigated by Bayesian program learning (Lake, Salakhutdinov, and Tenenbaum 2015), memory machines (Graves, Wayne, and Danihelka 2014; Santoro et al. 2016), meta learning (Finn, Abbeel, and Levine 2017; Yoon et al. 2018), metric learning (Qi, Brown, and Lowe 2018; Snell, Swersky, and Zemel 2017; Vinyals et al. 2016), etc. However, these approaches are designed for the standard classification task. Hence, they may lack the adaptation capacity for few-shot object detection.

3 Source Detection Transfer

To begin with, we formulate few-shot object detection in a practical transfer learning setting. **First**, we assume that, we can access to a published detection benchmark with C_s object categories. It is used as large-scale dataset for model pretraining in the source domain. **Second**, we aim at addressing few-shot detection in the target domain. Specifically, this task consists of C_t object categories. For each category, there are only N fully-annotated training images, e.g., N=5 for 5-shot case. **Finally**, we consider a challenging transfer scenario, i.e., object categories are non-overlapped between source and target domains, for evaluating whether our framework can generalize well on new object categories.

Detection Backbone. In this work, we choose the SSD-style detector (Liu et al. 2016; Liu, Huang, and others 2018) as backbone. One reason is that, multi-scale spatial receptive fields in this architecture provide rich contexts. Additionally, its concise detection design promotes flexibility of our transfer framework in practice. In particular, the SSD-style detector is a one-stage detection framework, which consists of detection heads on K spatial scales. For each spatial scale, the detection heads contain bounding box regressor (BBOX) and object+background classifier (OBJ+BG).

Source Detection Transfer. To generalize few-shot learning in the target domain, we first pretrain the SSD-style detector with large-scale benchmark in the source domain. In the following, we explain how to transfer source-domain detection heads (i.e., BBOX and OBJ+BG), so that one can leverage prior knowledge as much as possible to reduce overfitting for few-shot object detection.

(1) Source BBOX: Fine-Tuning. BBOX is used for localization. As it is shared among different categories, source-domain BBOX can be reused in the target domain. Furthermore, source-domain BBOX is pretrained with rich annotations in the large-scale dataset. Hence, fine-tuning this BBOX is often reliable to localize new objects, even though we only have a few training images in the target domain.

(2) Source BG: Fine-Tuning. OBJ+BG is used for classification. In this work, we factorize OBJ+BG separately into OBJ and BG classifiers. The reason is that, BG is a binary classifier (object or background), i.e., it is shared among different object categories. In this case, the pretrained BG can be reused in the target domain by fine-tuning.

(3) Source OBJ: Preserving. The last but the most challenging head is OBJ, i.e., multi-object classifier. Note that, object categories in the target domain are non-overlapped with those in the source domain. Traditionally, one should unload source-domain OBJ and add a new target-domain OBJ. However, adding new OBJ directly on top of high-dimensional feature would introduce a large number of randomly-initialized parameters, especially for multi-scale design in SSD-style frameworks. As a result, it is often hard to train such new OBJ from scratch, when we have only a few annotated images in the target domain. Alternatively, we propose to preserve source-domain OBJ and add a new target-domain OBJ on top of it. The main reason is that, the dimensionality of prediction score in source-domain OBJ is often much smaller than the number of feature channels in convolutional layers. When adding a new target-domain OBJ on top of source-domain OBJ, we will introduce fewer extra parameters and therefore alleviate overfitting.

Context-Transformer Between Source and Target OBJs. To some degree, preserving source-domain OBJ can reduce the training difficulty of target-domain OBJ. However, simple source detection transfer is not enough to address the underlying problem of few-shot object detection, i.e., object confusion introduced by annotation scarcity in the target domain. Hence, it is still necessary to further exploit target-domain knowledge effectively from only a few annotated training images. As mentioned in our introduction, humans often leverage contexts as a discriminative clue to distinguish such few-shot confusion. Motivated by this, we embed a novel Context-Transformer between source and target OBJs. It can automatically exploit contexts, with the guidance of source domain object knowledge from source OBJ. Then, it can integrate such relational clues to enhance target OBJ for few-shot detection.

4 Context-Transformer

In this section, we introduce Context-Transformer for few-shot object detection. Specifically, it is a novel plug-and-play module between source and target OBJs. We name it as Context-Transformer, because it consists of two submodules to transform contexts, i.e., affinity discovery and context aggregation. The whole framework is shown in Fig. 2.

4.1 Affinity Discovery

In SSD-style detectors (Liu et al. 2016), prior boxes are default anchor boxes with various aspect ratios. Since classification is performed over the representations of these boxes, affinity discovery first constructs a set of contextual fields for prior boxes. Subsequently, it exploits relations between prior boxes and contextual fields in a target-domain image, with guidance of source-domain object knowledge.

Source-Domain Object Knowledge of Prior Boxes. For a target-domain image, we should first find reliable representations of prior boxes, in order to perform affinity discovery under few-shot settings. Specifically, we feed a target-domain image into the pretrained SSD-style detector, and

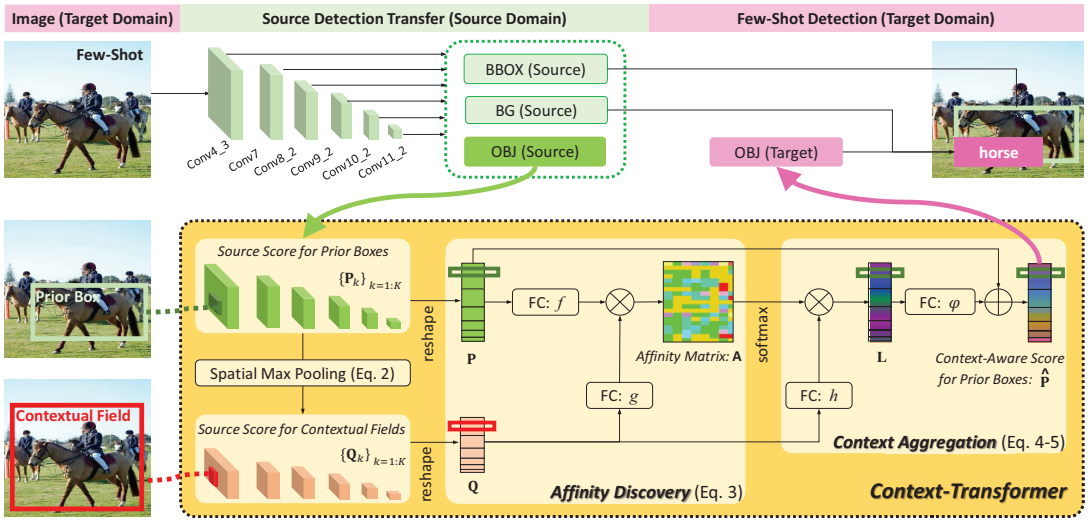

Figure 2: Few-Shot Detection with Context-Transformer. It is a plug-and-play module between source and target OBJs, based on SSD-style detectors. It consists of affinity discovery and context aggregation, which can effectively reduce object confusion in the few-shot target domain, by exploiting contexts in a concise transfer framework. More details can be found in Section 4.

extract the score tensor from source OBJ (before softmax),

$$\mathbf{P}_k \in \mathbb{R}^{H_k \times W_k \times (M_k \times C_s)}, \quad k = 1, ..., K, \tag{1}$$

where $\mathbf{P}_k(h, w, m, :) \in \mathbb{R}^{C_s}$ is a source-domain score vector, w.r.t., the prior box with the m-th aspect ratio located at (h, w) of the k-th spatial scale. We would like to emphasize that, the score of source-domain classifier often provides rich semantic knowledge about target-domain object categories (Tzeng et al. 2015; Yim et al. 2017). Hence, $\{\mathbf{P}_k\}_{k=1}^K$ is a preferable representation of prior boxes for a target-domain image. Relevant visualization can be found in our supplementary material.

Contextual Field Construction via Pooling. After obtaining the representation of prior boxes, we construct a set of contextual fields for comparison. Ideally, we hope that contextual fields are not constructed with excessive spatial assumptions and complicated operations, due to the fact that we only have a few training images on hand. A naive strategy is to use all prior boxes directly as contextual fields. However, there are approximately 10,000 prior boxes in the SSD-style architecture. Comparing each prior box with all others would apparently introduce unnecessary learning difficulty for few-shot cases. Alternatively, humans often check sparse contextual fields, instead of paying attention to every tiny detail in an image. Motivated by this observation, we propose to perform spatial pooling (e.g., max pooling) over prior boxes $\mathbf{P}_k$. As a result, we obtain the score tensor $\mathbf{Q}_k \in \mathbb{R}^{U_k \times V_k \times (M_k \times C_s)}$ for a set of contextual fields,

$$\mathbf{Q}_k = SpatialPool(\mathbf{P}_k), \quad k = 1, ..., K, \tag{2}$$

where $U_k \times V_k$ is the size of the k-th scale after pooling.

Affinity Discovery. To discover affinity between prior boxes and contextual fields, we compare them according to their source-domain scores. For convenience, we reshape score tensors $\mathbf{P}_{1:K}$ and $\mathbf{Q}_{1:K}$ respectively as matrices $\mathbf{P} \in \mathbb{R}^{D_p \times C_s}$ and $\mathbf{Q} \in \mathbb{R}^{D_q \times C_s}$, where each row of $\mathbf{P}$ (or $\mathbf{Q}$) refers to the source-domain score vector of a prior box (or a contextual field). Moreover, $D_p = \sum_{k=1}^K H_k \times W_k \times M_k$ and $D_q = \sum_{k=1}^K U_k \times V_k \times M_k$ are respectively the total number of prior boxes and contextual fields in a target-domain image. For simplicity, we choose the widely-used dot-product kernel to compare $\mathbf{P}$ and $\mathbf{Q}$ in the embedding space. As a result, we obtain an affinity matrix $\mathbf{A} \in \mathbb{R}^{D_p \times D_q}$ between prior boxes and contextual fields,

$$\mathbf{A} = f(\mathbf{P}) \times g(\mathbf{Q})^\top, \tag{3}$$

where $\mathbf{A}(i, :) \in \mathbb{R}^{1 \times D_q}$ indicates the importance of all contextual fields, w.r.t., the i-th prior box. $f(\mathbf{P}) \in \mathbb{R}^{D_p \times C_s}$ and $g(\mathbf{Q}) \in \mathbb{R}^{D_q \times C_s}$ are embeddings for prior boxes and contextual fields respectively, where f (or g) is a fully-connected layer that is shared among prior boxes (or contextual fields). These layers can increase learning flexibility of kernel computation. To sum up, affinity discovery allows a prior box to identify its important contextual fields automatically from various aspect ratios, locations and spatial scales. Such diversified relations provide discriminative clues to reduce object confusion caused by annotation scarcity.

4.2 Context Aggregation

After finding affinity between prior boxes and contextual fields, we use it as a relational attention to integrate contexts into the representation of each prior box.

Context Aggregation. We first add $softmax$ on each row of $\mathbf{A}$. In this case, $softmax(\mathbf{A}(i, :))$ becomes a gate vector that indicates how important each contextual field is for the i-th prior box. We use it to summarize all the contexts $\mathbf{Q}$ attentively,

$$\mathbf{L}(i, :) = softmax(\mathbf{A}(i, :)) \times h(\mathbf{Q}), \tag{4}$$

where $\mathbf{L}(i, :)$ is the weighted contextual vector for the i-th prior box (i=1, ..., D_p). Additionally, $h(\mathbf{Q}) \in \mathbb{R}^{D_q \times C_s}$

refers to a contextual embedding, where h is a fully-connected layer to promote learning flexibility. Finally, we aggregate the weighted contextual matrix $\mathbf{L} \in \mathbb{R}^{D_p \times C_s}$ into the original score matrix $\mathbf{P}$, and obtain the context-aware score matrix of prior boxes $\widehat{\mathbf{P}} \in \mathbb{R}^{D_p \times C_s}$,

$$\widehat{\mathbf{P}} = \mathbf{P} + \varphi(\mathbf{L}). \tag{5}$$

Similarly, the embedding $\varphi(\mathbf{L}) \in \mathbb{R}^{D_p \times C_s}$ is constructed by a fully-connected layer φ. Since $\widehat{\mathbf{P}}$ is context-aware, we expect that it can enhance the discriminative power of prior boxes to reduce object confusion in few-shot detection.

Target OBJ. Finally, we feed $\widehat{\mathbf{P}}$ into target-domain OBJ,

$$\widehat{\mathbf{Y}} = softmax(\widehat{\mathbf{P}} \times \Theta), \tag{6}$$

where $\widehat{\mathbf{Y}} \in \mathbb{R}^{D_p \times C_t}$ is the target-domain score matrix for classification. Note that, target OBJ is shared among different aspect ratios and spatial scales in our design, with a common parameter matrix $\Theta \in \mathbb{R}^{C_s \times C_t}$. One reason is that, each prior box has combined with its vital contextual fields of different aspect ratios and spatial scales, i.e., each row of $\widehat{\mathbf{P}}$ has become a multi-scale score vector. Hence, it is unnecessary to assign an exclusive OBJ on each individual scale. More importantly, target domain is few-shot. The shared OBJ can effectively reduce overfitting in this case.

Discussions. We further clarify the differences between related works and our Context-Transformer. **(1) Few-Shot Learners vs. Context-Transformer**. Few-shot learners (Snell, Swersky, and Zemel 2017; Finn, Abbeel, and Levine 2017; Qi, Brown, and Lowe 2018) and our Context-Transformer follow the spirit of learning with little supervision, in order to effectively generalize model based on few training samples. However, most few-shot learners are designed for standard classification tasks. Hence, they are often used as a general classifier without taking any detection insights into account. On the contrary, our Context-Transformer is a plug-and-play module for object detection. Via exploiting contexts in a concise transfer framework, it can adaptively generalize source-domain detector to reduce object confusion in the few-shot target domain. In fact, our experiment shows that, one can fine-tune the pretrained detector with Context-Transformer in the training phase, and flexibly unload it in the testing phase without much loss of generalization. All these facts make Context-Transformer a preferable choice for few-shot detection. **(2) Non-local Transformer vs. Context-Transformer**. Non-local Transformer (Wang et al. 2018) and our Context-Transformer follow the spirit of attention (Vaswani et al. 2017) for modeling relations. However, the following differences make our Context-Transformer a distinct module. First, Non-local Transformer does not take any few-shot insights into account. Hence, it would not be helpful to reduce training difficulty with little supervision. Alternatively, our Context-Transformer leverages source knowledge as guidance to alleviate overfitting in few-shot cases. Second, Non-local Transformer is not particularly designed for object detection. It is simply embedded between two convolution blocks in the standard CNN for space/spacetime modeling. Alternatively, our Context-Transformer is developed for few-shot object detection. We elaborately embed it between source and target OBJs in a SSD-style detection framework, so that it can tackle object confusions caused by annotation scarcity. Third, Non-local Transformer is a self-attention module, which aims at learning space/spacetime dependencies in general. Alternatively, our Context-Transformer is an attention module operated between prior boxes and contextual fields. It is used to automatically discover important contextual fields for each prior box, and subsequently aggregate such affinity to enhance OBJ for few-shot detection. In our experiments, we compare our Context-Transformer with these related works to show effectiveness and advancement.

5 Experiments

To evaluate our approach effectively, we adapt the popular benchmarks as two challenging settings, i.e., few-shot object detection, and incremental few-shot object detection. More results can be found in our supplementary material.

5.1 Few-Shot Object Detection

Data Settings. First, we set VOC07+12 as our target-domain task. The few-shot training set consists of N images (per category) that are randomly sampled from the original train/val set. Unless stated otherwise, N is 5 in our experiments. Second, we choose a source-domain benchmark for pretraining. To evaluate the performance of detecting novel categories in the target domain, we remove 20 categories of COCO that are overlapped with VOC, and use the rest 60 categories of COCO as source-domain data. Finally, we report the results on the official test set of VOC2007, by mean average precision (mAP) at 0.5 IoU threshold.

Implementation Details. We choose a recent SSD-style detector (Liu, Huang, and others 2018) as basic architecture, which is built upon 6 spatial scales (i.e., 38×38, 19×19, 10×10, 5×5, 3×3, 1×1). For contextual filed construction, we perform spatial max pooling on the first 4 scales of source-domain score tensors, where the kernel sizes are 3, 2, 2, 2 and the stride is the same as the kernel size. The embedding functions in Context-Transformer are residual-style FC layers, where input and output have the same number of channels. Finally, we implement our approach with PyTorch (Paszke et al. 2017), where all the experiments run on 4 TitanXp GPUs. For pre-training in the source domain, we follow the details of original SSD-style detectors (Liu, Huang, and others 2018; Liu et al. 2016). For fine-tuning in the target domain, we set the implementation details where the batch size is 64, the optimization is SGD with momentum 0.9, the initial learning rate is 4×10^{-3} (decreased by 10 after 3k and 3.5k iterations), the weight decay is 5×10^{-4}, the total number of training iterations is 4k.

Source Detection Transfer. The key design in Source Detection Transfer is OBJ. To reduce object confusion in the few-shot target domain, we propose to preserve source OBJ, and embed Context-Transformer between source and target OBJs. In Table 1, we evaluate the effectiveness of this design, by comparison with baseline (i.e., traditional fine-tuning with only target OBJ). First, our approach outperforms baseline, by adding target OBJ on

Method	OBJ (S)	Context-Transformer	OBJ (T)	mAP
Baseline	×	×	√	39.4
Ours	√	×	√	40.9
	×	√	√	41.5
	√	√	√	**43.8**
	√	√ → ×	√	43.4

Table 1: Source Detection Transfer. Baseline: traditional fine-tuning with target-domain OBJ. √ → ×: We fine-tune the pretrained detector with Context-Transformer in the training phase, and then unload it in the testing phase.

Context Construction	mAP	Embedding Layer	mAP
Without	42.5	*Without*	42.2
Pool_avg	43.5	*FC_no residual*	43.0
Pool_max	**43.8**	*FC_residual*	**43.8**
Affinity Discovery	mAP	OBJ (Target)	mAP
Euclidean	43.5	*Separate*	41.4
Cosine	**43.8**	*Share*	**43.8**

Table 2: Designs of Context-Transformer.

top of source OBJ. It shows that, preserving source OBJ can alleviate overfitting for few-shot learning. Second, our approach outperforms baseline, by adding target OBJ on top of Context-Transformer. It shows that, Context-Transformer can effectively reduce confusion by context learning. Third, our approach achieves the best when we embed Context-Transformer between source and target OBJs. In this case, Context-Transformer can sufficiently leverage source-domain knowledge to enhance target OBJ. Note that, our design only introduces 15.6K extra parameters (i.e., Context-Transformer: 14.4K, target OBJ: 1.2K), while baseline introduces 2,860K extra parameters by adding target OBJ directly on top of multi-scale convolution features. It shows that, our design is of high efficiency. Finally, we unload Context-Transformer in the testing, after applying it in the training. As expected, the performance drops marginally, indicating that Context-Transformer gradually generalizes few-shot detector by learning contexts during training.

Designs of Context-Transformer. We investigate key designs of Context-Transformer in Table 2. (1) Context Construction. First, the pooling cases are better. It shows that, we do not need to put efforts on every tiny details in the images. Pooling can effectively reduce the number of contextual fields, and consequently alleviate learning difficulty in comparison between prior boxes and contexts. Additionally, max pooling is slightly better than average pooling. Hence, we choose max pooling. (2) Embedding Functions. As expected, Context-Transformer performs better with FC layers, due to the improvement of learning flexibility. Additionally, the residual style is slightly better than the no-residual case, since it can reduce the risk of random initialization especially for few-shot learning. Hence, we choose the residual-style FC layers. (3) Affinity Discovery. We compute affinity by two popular similarity metric, i.e., Cosine (dot-product) and Euclidean distance. The results are comparable, showing that Context-Transformer is robust to the metric choice. For simplicity, we choose Cosine in our paper. (4) OBJ (Target). Sharing OBJ (Target) among spatial

No. of Shots (N)	1	2	3	5	10	all
Baseline	21.5	27.9	33.5	39.4	49.2	80.7
Ours	**27.0**	**30.6**	**36.8**	**43.8**	**51.4**	**81.5**

Table 3: Influence of Training Shots.

Trials No.	1	2	3	4	5	mean±std
Baseline	41.7	43.1	37.9	43.5	38.2	(Baseline)
Ours	**43.7**	**46.5**	**41.5**	**45.7**	**41.1**	40.4±2.0
Trials No.	6	7	8	9	10	mean±std
Baseline	40.4	38.8	40.7	38.7	40.6	(Ours)
Ours	**44.7**	**41.9**	**43.4**	**42.4**	**42.3**	**43.3±1.8**

Table 4: Influence of Random Trials (5-Shot Case).

SSD-Style Framework	(Liu et al. 2016)		(Liu et al. 2018)	
	Baseline	Ours	Baseline	Ours
mAP	35.3	**38.7**	39.4	**43.8**

Table 5: Influence of SSD-Style Framework.

scales achieves better performance. This perfectly matches our insight in Section 4.2, i.e., each prior box has combined with its key contextual fields of various spatial scales, after learning with Context-Transformer. Hence, it is unnecessary to assign an exclusive OBJ (Target) for each scale separately.

Influence of Shot and Framework. First, the detection performance tends to be improved as the number of training shots increases in Table 3. Interestingly, we find that the margin between our approach and baseline tends to decline gradually, when we have more training shots. This matches our insight that, Context-Transformer is preferable to distinguish object confusion caused by low data diversity. When the number of training samples increases in the target domain, such few-shot confusion would be alleviated with richer annotations. But still, Context-Transformer can model discriminative relations to boost detection in general. Hence, our approach also outperforms baseline for all-shot setting. Second, our approach exhibits high robustness to random trials (Table 4), where we run our approach on extra 10 random trials for 5-shot case. The results show that our approach consistantly outperforms baseline. Finally, we build Context-Transformer upon two SSD-style frameworks (Liu et al. 2016) and (Liu, Huang, and others 2018). In Table 5, our approach significantly outperforms baseline. The result is better on (Liu, Huang, and others 2018) due to multi-scale dilation.

Comparison with Related Learners. We compare Context-Transformer with popular few-shot learners (Snell, Swersky, and Zemel 2017; Qi, Brown, and Lowe 2018) and Non-local Transformer (Wang et al. 2018). We re-implement these approaches in our transfer framework, where we replace Context-Transformer with these learners. More implementation details can be found in our supplementary material. In Fig. 3, Context-Transformer outperforms Prototype (Snell, Swersky, and Zemel 2017) and Imprinted (Qi, Brown, and Lowe 2018), which are two well-known few-shot classifiers. It shows that, general methods may not be sufficient for few-shot detection. Furthermore, Context-Transformer outperforms Non-local (Wang et al. 2018). It shows that, it is preferable to discover affinity between prior

VOC2007 (5-Shot Case)	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	avg
Prototype (Snell et al. 2017)	50.0	55.0	23.7	**26.1**	8.9	54.2	71.2	41.6	29.8	23.6	34.0	30.7	46.3	53.3	60.2	**21.9**	37.3	24.3	50.2	**54.4**	39.8
Imprinted (Qi et al. 2018)	49.1	54.8	26.0	23.5	14.7	53.0	71.2	53.0	30.4	21.0	34.0	28.6	48.1	**56.4**	63.5	21.4	39.2	32.9	45.1	50.8	40.9
Non-local (Wang et al. 2018)	51.9	58.1	25.3	**26.1**	8.5	49.7	71.9	55.3	**32.3**	20.1	31.9	**32.1**	44.7	55.8	63.7	16.2	41.7	**33.2**	52.4	49.9	41.0
Our Context-Transformer	**55.4**	**59.1**	**28.6**	23.9	**15.9**	**58.3**	**74.5**	**57.1**	31.4	**26.0**	**38.1**	31.7	**55.8**	56.1	**64.1**	18.1	**45.8**	33.0	**53.2**	49.9	**43.8**

Figure 3: Comparison with Related Learners (Few-Shot Object Detection). We re-implement these learners in our transfer framework, where we replace our Context-Transformer by them.

Before Incremental		Split1		Split2		Split3	
		S	T	S	T	S	T
S (All)	Shmelkov2017	67.2	-	68.3	-	69.3	-
	Kang2019	69.7	-	72.0	-	70.8	-
	Ours	**72.9**	-	**73.0**	-	**74.0**	-
After Incremental		**Split1**		**Split2**		**Split3**	
		S	T	S	T	S	T
S+T (1Shot)	Shmelkov2017	52.5	23.9	54.0	19.2	54.6	21.4
	Kang2019	66.4	14.8	**68.2**	15.7	65.9	19.2
	Ours	**67.4**	**34.2**	68.1	**26.0**	**66.8**	**29.3**
S+T (5Shot)	Shmelkov2017	57.4	38.8	58.1	32.5	59.1	31.8
	Kang2019	63.4	33.9	66.6	30.1	64.6	40.6
	Ours	**67.3**	**44.2**	**67.4**	**36.3**	**67.4**	**40.8**

Table 6: Incremental Few-Shot Object Detection (mAP). S: source-domain classes. T: target-domain classes. More details can be found in Section 5.2.

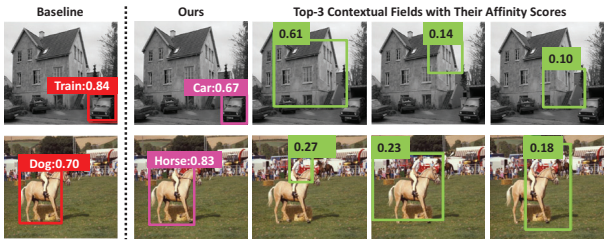

Figure 4: Context Affinity. Context-Transformer can distinguish a *car* from a *train*, when it finds that there is a family house (1st context) with windows (2nd context) and entrance stairs (3rd context). Similarly, it can distinguish a *horse* from a *dog*, when it finds that there is a person (1st context) on top of this animal (2nd and 3rd contexts).

boxes and contextual fields to reduce object confusion, instead of self-attention among prior boxes.

5.2 Incremental Few-Shot Object Detection

In practice, many applications refer to an incremental scenario (Kang et al. 2019), i.e., the proposed framework should boost few-shot detection in a new target domain, while maintaining the detection performance in the previous source domain. Our transfer framework can be straightforwardly extended to address it. Specifically, we add a residual-style FC layer on top of the pretrained source-domain OBJ, which allows us to construct a new source-domain OBJ that can be more compatible with target-domain OBJ. Then, we concatenate new source and target OBJs to classify objects in both domains. More implementation details can be found in our supplementary material.

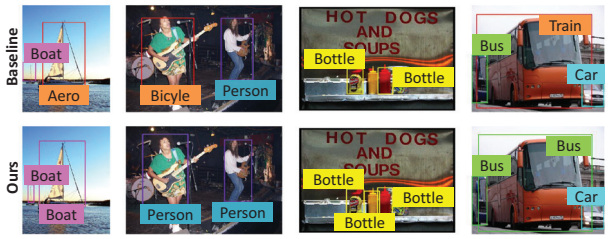

Figure 5: Detection Visualization (5-Shot Case).

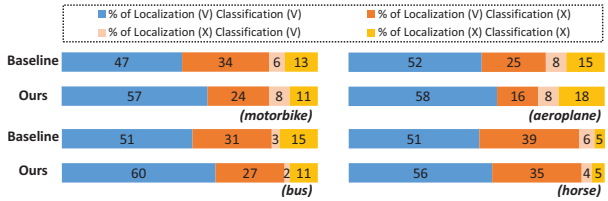

Figure 6: Object Confusion (Top-4 Improvement). Context-Transformer can effectively reduce the correctly-localized but wrongly-classified instances (e.g., baseline vs. ours: 34% vs. 24% for *motorbike*), and promote the correct detection (e.g., baseline vs. ours: 47% vs. 57% for *motorbike*).

To evaluate this incremental scenario, we follow the original data settings of (Kang et al. 2019). There are 3 data splits that are built upon VOC07+12 (train/val) and VOC07 (test). For each split, there are 15 source-domain classes (base) and 5 target-domain classes (novel). For each target-domain class, there are only N-shot annotated bounding boxes. We compare our approach with two recent incremental detection approaches, i.e., (Shmelkov, Schmid, and Alahari 2017) and (Kang et al. 2019) As shown in Table 6, our approach can effectively boost few-shot detection in the target domain as well as maintain the performance in the source domain, and significantly outperform the state-of-the-art approaches.

5.3 Visualization

Context Affinity. In Fig. 4, we show top-3 important contextual fields learned by our Context-Transformer. As we can see, context affinity can correct object confusion to boost few-shot detection. Furthermore, the sum of top-3 affinity scores is over 0.6. It illustrates that, Context-Transformer can learn to exploit sparse contextual fields for a prior box, instead of focusing on every tiny detail in the image.

Detection Visualization. In Fig. 5, we show the detection performance for 5-shot case. One can clearly see that, our approach correctly detects different objects, while vanilla

fine-tuning introduces large object confusions.

Object Confusion Analysis. In Fig. 6, we show top-4 improved categories by our approach. As expected, Context-Transformer can largely reduce object confusion and boost performance in few-shot cases.

6 Conclusion

In this work, we propose a Context-Transformer for few-shot object detection. By attentively exploiting multi-scale contextual fields within a concise transfer framework, it can effectively distinguish object confusion caused by annotation scarcity. The extensive results demonstrate the effectiveness of our approach.

7 Acknowledgements

This work is partially supported by the National Key Research and Development Program of China (No. 2016YFC1400704), and National Natural Science Foundation of China (61876176, U1713208), Shenzhen Basic Research Program (JCYJ20170818164704758, CXB201104220032A), the Joint Lab of CAS-HK, Shenzhen Institute of Artificial Intelligence and Robotics for Society.

References

Bell, S.; Lawrence Zitnick, C.; Bala, K.; and Girshick, R. 2016. Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks. In *CVPR*, 2874–2883.

Bilen, H., and Vedaldi, A. 2016. Weakly supervised deep detection networks. In *CVPR*, 2846–2854.

Chen, X., and Gupta, A. 2017. Spatial memory for context reasoning in object detection. In *ICCV*, 4086–4096.

Dong, X.; Zheng, L.; Ma, F.; Yang, Y.; and Meng, D. 2018. Few-example object detection with model communication. *IEEE-TPAMI* 41(7):1641–1654.

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, 1126–1135. JMLR. org.

Girshick, R.; Donahue, J.; Darrell, T.; and Malik, J. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 580–587.

Girshick, R. 2015. Fast r-cnn. In *ICCV*, 1440–1448.

Graves, A.; Wayne, G.; and Danihelka, I. 2014. Neural turing machines. *arXiv preprint arXiv:1410.5401*.

He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask r-cnn. In *CVPR*, 2961–2969.

Hoffman, J.; Guadarrama, S.; Tzeng, E. S.; Hu, R.; Donahue, J.; Girshick, R.; Darrell, T.; and Saenko, K. 2014. Lsda: Large scale detection through adaptation. In *NIPS*, 3536–3544.

Hu, H.; Gu, J.; Zhang, Z.; Dai, J.; and Wei, Y. 2018. Relation networks for object detection. In *CVPR*, 3588–3597.

Kang, B.; Liu, Z.; Wang, X.; Yu, F.; Feng, J.; and Darrell, T. 2019. Few-shot object detection via feature reweighting. In *ICCV*, 8420–8429.

Kantorov, V.; Oquab, M.; Cho, M.; and Laptev, I. 2016. Contextlocnet: Context-aware deep network models for weakly supervised localization. In *ECCV*, 350–365. Springer.

Lai, B., and Gong, X. 2017. Saliency guided end-to-end learning forweakly supervised object detection. In *IJCAI*, 2053–2059. AAAI Press.

Lake, B. M.; Salakhutdinov, R.; and Tenenbaum, J. B. 2015. Human-level concept learning through probabilistic program induction. *Science* 350(6266):1332–1338.

Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; and Berg, A. C. 2016. Ssd: Single shot multibox detector. In *ECCV*, 21–37. Springer.

Liu, S.; Huang, D.; et al. 2018. Receptive field block net for accurate and fast object detection. In *ECCV*, 385–400.

Mottaghi, R.; Chen, X.; Liu, X.; Cho, N.-G.; Lee, S.-W.; Fidler, S.; Urtasun, R.; and Yuille, A. 2014. The role of context for object detection and semantic segmentation in the wild. In *CVPR*, 891–898.

Oliva, A., and Torralba, A. 2007. The role of context in object recognition. *Trends in cognitive sciences* 11(12):520–527.

Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; and Lerer, A. 2017. Automatic differentiation in pytorch.

Qi, H.; Brown, M.; and Lowe, D. G. 2018. Low-shot learning with imprinted weights. In *CVPR*, 5822–5830.

Redmon, J.; Divvala, S.; Girshick, R.; and Farhadi, A. 2016. You only look once: Unified, real-time object detection. In *CVPR*, 779–788.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 91–99.

Santoro, A.; Bartunov, S.; Botvinick, M.; Wierstra, D.; and Lillicrap, T. 2016. Meta-learning with memory-augmented neural networks. In *ICML*, 1842–1850.

Shmelkov, K.; Schmid, C.; and Alahari, K. 2017. Incremental learning of object detectors without catastrophic forgetting. In *ICCV*, 3400–3409.

Snell, J.; Swersky, K.; and Zemel, R. 2017. Prototypical networks for few-shot learning. In *NIPS*, 4077–4087.

Tang, Y.; Wang, J.; Gao, B.; Dellandréa, E.; Gaizauskas, R.; and Chen, L. 2016. Large scale semi-supervised object detection using visual and semantic knowledge transfer. In *CVPR*, 2119–2128.

Tang, P.; Wang, X.; Bai, X.; and Liu, W. 2017. Multiple instance detection network with online instance classifier refinement. In *CVPR*, 2843–2851.

Tzeng, E.; Hoffman, J.; Darrell, T.; and Saenko, K. 2015. Simultaneous deep transfer across domains and tasks. In *ICCV*, 4068–4076.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *NIPS*, 5998–6008.

Vinyals, O.; Blundell, C.; Lillicrap, T.; Wierstra, D.; et al. 2016. Matching networks for one shot learning. In *NIPS*, 3630–3638.

Wang, X.; Girshick, R.; Gupta, A.; and He, K. 2018. Non-local neural networks. In *CVPR*, 7794–7803.

Yim, J.; Joo, D.; Bae, J.; and Kim, J. 2017. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *CVPR*, 4133–4141.

Yoon, J.; Kim, T.; Dia, O.; Kim, S.; Bengio, Y.; and Ahn, S. 2018. Bayesian model-agnostic meta-learning. In *NIPS*, 7332–7342.