Strong-Weak Distribution Alignment for Adaptive Object Detection

Kuniaki Saito[1], Yoshitaka Ushiku[2], Tatsuya Harada[2,3] and Kate Saenko[1]

[1]Boston University, [2]The University of Tokyo, [3]RIKEN
{keisaito, saenko}@bu.edu, {ushiku, harada}@mi.t.u-tokyo.ac.jp

Abstract

We propose an approach for unsupervised adaptation of object detectors from label-rich to label-poor domains which can significantly reduce annotation costs associated with detection. Recently, approaches that align distributions of source and target images using an adversarial loss have been proven effective for adapting object classifiers. However, for object detection, fully matching the entire distributions of source and target images to each other at the global image level may fail, as domains could have distinct scene layouts and different combinations of objects. On the other hand, strong matching of local features such as texture and color makes sense, as it does not change category level semantics. This motivates us to propose a novel method for detector adaptation based on strong local alignment and weak global alignment. Our key contribution is the weak alignment model, which focuses the adversarial alignment loss on images that are globally similar and puts less emphasis on aligning images that are globally dissimilar. Additionally, we design the strong domain alignment model to only look at local receptive fields of the feature map. We empirically verify the effectiveness of our method on four datasets comprising both large and small domain shifts. Our code is available at https://github.com/VisionLearningGroup/DA_Detection.

1. Introduction

Deep convolutional neural networks have greatly improved object recognition accuracy [17], but remain reliant on large quantities of labeled training data. For object detection, annotation is particularly burdensome: each instance of an object category in every image must be annotated with a precise bounding box. Transferring pre-trained models from label-rich domains is an attractive solution, but dataset bias often reduces their generalization to novel data [30].

Various methods for unsupervised domain adaptation (UDA) have been proposed to tackle the dataset bias problem [10, 40, 39, 24], most of which are based on domain-invariant alignment of the feature [31] or image [20, 14] distributions. Recent methods align the source and target distributions of examples using adversarial learning and are motivated by theoretical results that bound the generalization error partially by the size of the discrepancy between domains [2, 1]. The conventional wisdom is therefore that

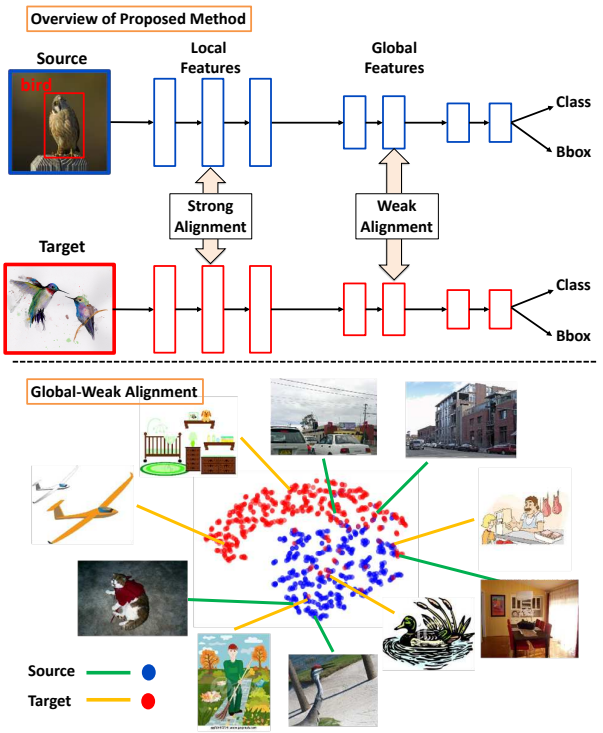


Figure 1. Upper: Our Strong-Weak model learns domain-invariant features that are strongly aligned at the local patch level and weakly (partially) aligned at the global scene level. Lower: Global features obtained by our proposed weak alignment method on Pascal to Clipart. The target features are partially aligned with source, which improves detection performance, as shown in our experiments.

discrepancy must be reduced at all costs, which can only be done if one fully aligns the distributions. In this paper, we argue that such *strong* domain alignment is only reasonable in closed problems, such as object classification settings where the source and target examples share the same categories and prior label distributions. In settings such as open-set classification [4, 33] or partial domain adaptation [41], strong alignment can be infeasible and could actually hurt performance.

In object detection this is particularly evident, as aligning global (image-level) features means that not only the object categories, but also backgrounds and scene layouts must be similar across domains. Yet this is precisely what the current state-of-the-art UDA method for detection, Adaptive Faster RCNN [5], attempts to do. It trains Faster RCNN with a domain classifier trained to distinguish source and target examples, while the feature extractor learns to deceive the domain classifier. Feature alignment is done both at the global image scale and at the instance (object) scale.

While the global matching might work well for small domain shifts that only affect the appearance/texture of objects (e.g. weather related shifts), it is likely to hurt performance for larger shifts that affect the layout of the scene, the number of objects and/or their co-occurrence. For example, source images may contain single objects, while target images may contain multiple smaller objects. Forcing invariance to such *global* features can hurt performance.On the other hand, strong alignment of *local* features would match the texture or color of the domains and should improve performance in most cases, because it will not change the category information but is likely to reduce the domain gap. In this paper, by "local" scale we do not mean the instance (object) scale but rather texture or color features with small receptive fields.

Motivated by these observations, we propose an unsupervised adaptation method for object detection that combines *weak global* alignment with *strong local* alignment, called the Strong-Weak Domain Alignment model (top of Fig. 1). We propose to apply *weak* alignment to the *global* features, partially aligning them to reduce the domain gap without hurting the performance of the model. We show an example of weak global alignment in the bottom of Fig. 1, where only the target images which contain one object are aligned with the source. Our key contribution is the weak global alignment model, which focuses the adversarial alignment loss toward images that are globally similar, and away from images that are globally dissimilar. Additionally, we achieve strong local alignment by constructing a domain classifier designed to look only at local features and to strictly align them with the other domain. We verify the effectiveness of our method in adaptation between both similar and dissimilar domains.

2. Related Work

Object Detection. The development of deep convolutional neural networks has boosted the performance of object detection. Having a strong backbone feature extractor is key for accurate detection models. Current detection networks can be categorized into two types: two-stage and one-stage. Faster-RCNN (FRCNN) [29] is a representative two-stage detector that generates coarse object proposals using region proposal networks (RPN) as the first stage, and feeds the proposals and cropped features into a classification module as the second stage. In this paper, we use the FRCNN as a base detector, however, our method should be applicable to other two-stage detectors and one-stage detectors such as YOLO [28] or SSD [21]. Detector back-bone networks are usually pre-trained on ImageNet [7] and need to be fine-tuned again with a large number of annotated object bounding boxes. Various datasets have been publicized for this purpose [8, 7, 19]. To deal with the deficit in such large annotated datasets, weakly supervised and semi-supervised object detection has been proposed in the literature [38, 3]. Although cross-domain object detection and especially unsupervised cross-domain object detection can also help with this problem, as far as we know, there is only one work that has tackled the task of unsupervised domain transfer of deep object detectors [5]. In this work, the feature alignment at the instance (object) scale was done for features cropped by region proposals. To effectively conduct feature alignment, region proposals have to precisely localize objects of interest. However, this is difficult to do for the target domain as we are not given ground truth proposals. The feature alignment may therefore hurt the performance of the model as we show in our experiments, which is why we do not conduct instance scale alignment in our work.

Domain Adaptation. The problem of bridging a gap between domains has been investigated for various visual applications such as image classification and semantic segmentation [30, 40, 43, 35]. To solve the problem, a large number of methods utilize feature distribution matching between training and testing domains. The basic idea is to measure some type of distance between different domains' feature distributions and train a feature extractor to minimize that distance. Various ways of measuring the distance have been proposed [9, 40, 39, 22, 24, 32]. Motivated by a theoretical result [2, 1], various approaches utilize the domain classifier [9, 40, 39] to measure domain discrepancy. They train a domain classifier and feature extractor in an adversarial way, as done for training GANs [11]. Such methods are designed to strictly align the feature distribution of the target with that of the source. In addition, Long *et al*. designed a loss function of the domain classifier to fully match features between domains [23] for image classification.

In this paper, we instead propose a weak feature alignment model for global features, and use strong alignment

only at the local level to strictly align the style of images across domains. Some research on GANs and domain adaptive semantic segmentation has shown that regularizing the domain classifier with task-specific classification loss can stabilize the adversarial training [26, 35]. Motivated by this approach, we further propose a method to regularize the domain classifier by the detection loss on source examples.

3. Method

The architecture of our proposed Strong-Weak DA model is illustrated in Fig. 3. We extract global features just before the RPN and local features from lower layers, and perform *weak global* alignment in the high-level feature space and *strong local* alignment in the low-level feature space. We further propose to stabilize the training of domain classifiers with the detection loss (Sec. 3.3).

3.1. Weak Global Feature Alignment

We utilize a domain classifier to align the target features with the source for the global-level feature alignment. Easy-to-classify target examples are far from source examples in the feature space while hard-to-classify target examples are near the source as shown in the left of Fig. 2. Therefore, focusing on hard-to-classify examples should achieve a weak alignment between domains. We propose to train a domain classifier to ignore easy-to-classify examples while focusing on hard-to-classify examples with respect to the classification of the domain.

We have access to a labeled source image x^s and bounding boxes for each image y^s drawn from a set of annotated source images $\{X_s, Y_s\}$, as well as an unlabeled target image x^t drawn from unlabeled target images X_t. The global feature vector is extracted by F. The domain classifier, D_g, is trained to predict the domain of input global features. Our learning formulation optimizes F so that the features are discriminative for the primary task of object detection, but are uninformative for the task of domain classification. The domain-label d is 1 for the source and 0 for the target. The network R takes features from F and outputs bounding boxes with a class label. R includes the Region Proposal Network (RPN) and other modules in Faster RCNN. The objective of the detection loss is summarized as:

$$\mathcal{L}_{cls}(F, R) = -\frac{1}{n_s}\sum_{i=1}^{n_s}\mathcal{L}_{det}(R(F({x_i}^s)), {y_i}^s) \quad (1)$$

where we assume that $\mathcal{L}_{det}$ contains all losses for detection such as a classification loss and a bounding-box regression loss. n_s denotes the number of source examples.

In existing methods [5], the objective for domain classification is the cross-entropy loss. As shown in Fig. 2, the loss of the easy-to-classify examples, which have high probability, is not negligible in this cross-entropy loss. This indicates that D_g and F account for all examples in the training procedure. Therefore, F tries to match the entire feature

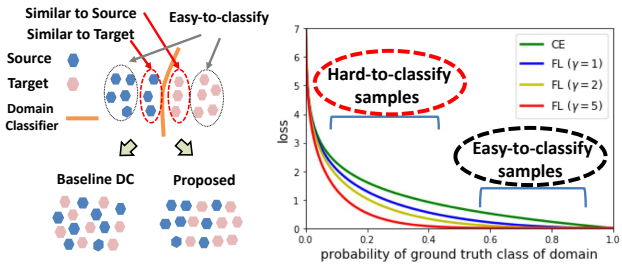


Figure 2. Left: Weak-distribution alignment using a domain classifier. Right: Standard cross-entropy loss and focal loss.

distribution, which is not desirable in domain adaptive object detection.

Instead, we want the domain classifier to ignore easy-to-classify examples while focusing on hard-to-classify examples. The problem with cross-entropy (CE) loss ($-\log p$) is that it puts non-negligible values of easy-to-classify examples where $p \in [0, 1]$ is the model's estimated probability for the class with label $d = 1$. We propose to add a modulating factor $f(p_t)$ to the cross-entropy loss, resulting in

$$-f(p_t)\log(p_t) \quad (2)$$

where we define p_t:

$$p_t = \begin{cases} p & \text{if } d = 1 \\ 1 - p & \text{otherwise.} \end{cases} \quad (3)$$

We choose a function that decreases as p_t increases. One example of such a loss function is Focal Loss (FL) [18]

$$\text{FL}(p_t) = -f(p_t)\log(p_t), f(p_t) = (1 - p_t)^\gamma \quad (4)$$

where γ controls the weight on hard-to-classify examples. FL is designed to put more weight on hard-to-classify examples than on easy ones during training, as shown in the right of Fig. 2. The feature extractor tries to deceive the domain classifier, that is, tries to increase the loss. However, the feature extractor cannot align the well-classified target examples with the source because the scale of gradients of such examples is very small. The same can be said about aligning source examples to the target. $f(p_t)$ can take other formulations if it satisfies the requirement described above. In experiments, we will show the result of another loss function that satisfies the condition. We denote the loss of the weak global-level domain classifier as $\mathcal{L}_{global}$ as follows,

$$\mathcal{L}_{global_s} = -\frac{1}{n_s}\sum_{i=1}^{n_s}(1 - D_g(F({x_i}^s))^\gamma \log(D_g(F({x_i}^s))) \quad (5)$$

$$\mathcal{L}_{global_t} = -\frac{1}{n_t}\sum_{i=1}^{n_t} D_g(F({x_i}^t))^\gamma \log(1 - D_g(F({x_i}^t))) \quad (6)$$

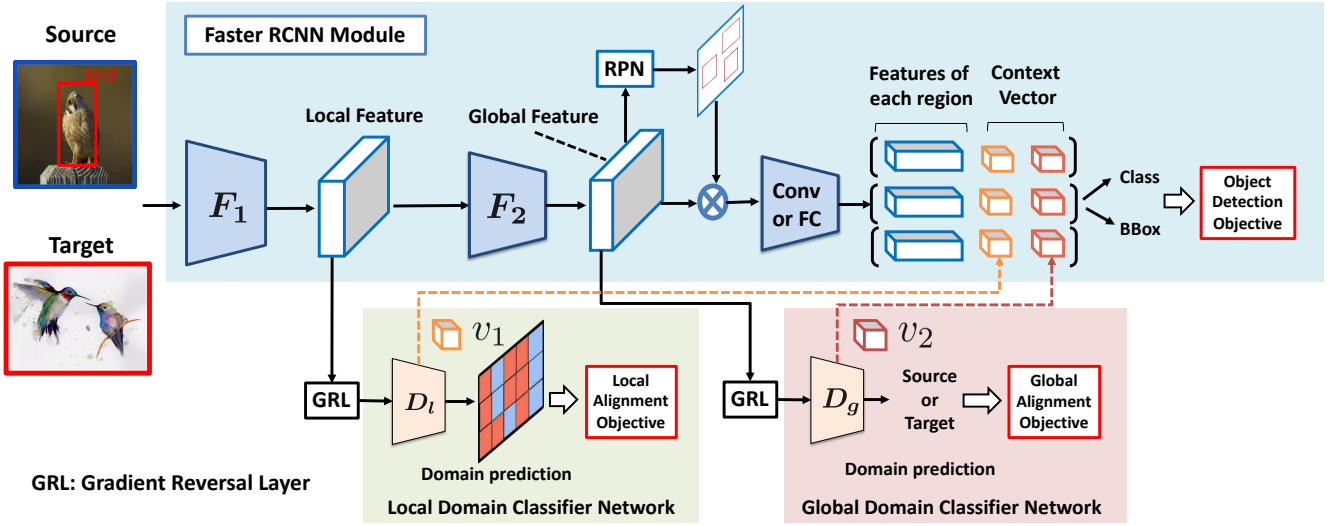


Figure 3. Proposed Network Architecture. Our method performs strong-local alignment by a local domain classifier network and weak-global alignment by a global domain classifier. The context vector is extracted by the domain classifiers and is concatenated in the layer before the final fully connected layer.

$$\mathcal{L}_{global}(F, D_g) = \frac{1}{2}(\mathcal{L}_{global_s} + \mathcal{L}_{global_t}) \quad (7)$$

where n_t denotes the number of target examples.

The gradients of this loss should change the parameters of low-level layers, which should also align low-level features, but the effect may not be strong enough. We thus propose to directly perform the alignment in local-level features in the next sub-section.

3.2. Strong Local Feature Alignment

The architecture of the local domain classifier, D_l, is designed to focus on the local features rather than global features. D_l is a fully-convolutional network with kernel-size equal to one. The feature extractor F is decomposed as $F_2 \circ F_1$ and the output of F_1 is the input to D_l as shown in Fig. 3. F_1 outputs a feature whose width and height is W and H respectively. D_l outputs a domain prediction map which has the same width and height as the input feature. We employed a least-squares loss to train the domain classifier following [25, 42]. This loss function stabilizes the training of the domain classifier and is empirically shown to be useful for aligning low-level features. The loss function of the strong local alignment $\mathcal{L}_{loc}$ is summarized as

$$\mathcal{L}_{loc_s} = \frac{1}{n_s HW} \sum_{i=1}^{n_s} \sum_{w=1}^{W} \sum_{h=1}^{H} D_l(F_1({x_i}^s))^2_{wh} \quad (8)$$

$$\mathcal{L}_{loc_t} = \frac{1}{n_t HW} \sum_{i=1}^{n_t} \sum_{w=1}^{W} \sum_{h=1}^{H} (1 - D_l(F_1({x_i}^t))_{wh})^2 \quad (9)$$

$$\mathcal{L}_{loc}(F, D_l) = \frac{1}{2}(\mathcal{L}_{loc_s} + \mathcal{L}_{loc_t}) \quad (10)$$

where $D_l(F_1({x_i}^s))_{wh}$ denotes the output of the domain classifier in each location. The loss is designed to align each receptive field of features with the other domain.

3.3. Context Vector based Regularization

We further propose a regularization technique to improve the performance of our model. As discussed above, regularizing the domain classifier with the segmentation loss was effective for stabilizing the adversarial training in domain adaptive segmentation [35]. The authors designed a domain classifier that outputs both the domain label and a semantic segmentation map. Motivated by this approach, we propose to stabilize the training of the domain classifier by the detection loss computed on source examples. We extract vectors v_1 and v_2 from the middle layers of the two domain classifiers respectively. These vectors should contain information about whole input image, which we call "context". Then, we concatenate the vectors with all region-wise features as shown in Fig. 3 and train the domain classifiers to minimize the detection loss on source examples as well as minimize domain classification loss. During the test phase, the vectors are forwarded to obtain outputs.

3.4. Overall Objective

We denote the objective of detection modules as $\mathcal{L}_{det}$, which contains the loss for region proposal networks and final classification and localization error. The adversarial loss $\mathcal{L}_{adv}(F, D)$ is summarized as,

$$\mathcal{L}_{adv}(F, D) = \mathcal{L}_{loc}(F_1, D_l) + \mathcal{L}_{global}(F, D_g) \quad (11)$$

Combined with the loss of detection on source examples, the overall objective is,

$$\max_{D} \min_{F,R} \mathcal{L}_{cls}(F, R) - \lambda \mathcal{L}_{adv}(F, D) \quad (12)$$

Table 1. Results on adpatation from PASCAL VOC to Clipart Dataset. Average precision (%) is evaluated on target images. G, I, CTX, L indicate global alignment, instance-level alignment, context-vector based regularization, and local-alignment respectively.

Method	G	I	CTX	L	aero	bcycle	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	hrs	bike	prsn	plnt	sheep	sofa	train	tv	MAP
Source Only					**35.6**	52.5	24.3	23.0	20.0	43.9	32.8	10.7	30.6	11.7	13.8	6.0	**36.8**	45.9	48.7	41.9	**16.5**	7.3	22.9	32.0	27.8
BDC-Faster	✓				20.2	46.4	20.4	19.3	18.7	41.3	26.5	6.4	33.2	11.7	**26.0**	1.7	36.6	41.5	37.7	44.5	10.6	20.4	33.3	15.5	25.6
DA-Faster	✓	✓			15.0	34.6	12.4	11.9	19.8	21.1	23.2	3.1	22.1	26.3	10.6	10.0	19.6	39.4	34.6	29.3	1.0	17.1	19.7	24.8	19.8
Proposed	✓				30.5	48.5	**33.6**	24.8	41.2	48.9	32.4	17.2	34.5	55.0	19.0	13.6	35.1	66.2	**63.0**	45.3	12.5	22.6	45.0	38.9	36.4
	✓		✓		31.7	**55.2**	30.9	26.8	**43.4**	47.5	**40.0**	7.9	**36.7**	50.0	14.3	**18.0**	29.2	**68.1**	62.3	50.4	13.4	24.5	**54.2**	45.8	37.5
	✓		✓	✓	26.2	48.5	32.6	**33.7**	38.5	**54.3**	37.1	**18.6**	34.8	**58.3**	17.0	12.5	33.8	65.5	61.6	**52.0**	9.3	**24.9**	54.1	**49.1**	**38.1**

Table 2. Results on PASCAL VOC in adaptation from PASCAL VOC to Clipart Dataset. Average precision (%) is evaluated on PASCAL. Our method does not degrade the performance on the source whereas BDC-Faster and DC-Faster degrade it.

Method	G	I	CTX	L	MAP
Source Only					77.5
BDC-Faster	✓				73.6
DA-Faster	✓	✓			66.4
Proposed	✓				**78.0**
	✓		✓		77.6
	✓		✓	✓	77.0

where λ controls the trade-off between detection loss and adversarial training loss. The sign of gradients is flipped by a gradient reversal layer proposed by [9]. Each mini-batch has one labeled source and one unlabeled target example.

4. Experiments

We evaluate our approach on four domain shifts–PASCAL [8] to Clipart [15], PASCAL to Watercolor [15], Cityscapes [6] to FoggyCityscapes [34], and GTA [16] to Cityscapes–to demonstrate that it is effective for adaptation between both dissimilar and similar domains. Additionally, we provide experiments to verify our claim that complete feature matching can degrade the performance of the model in the target domain.

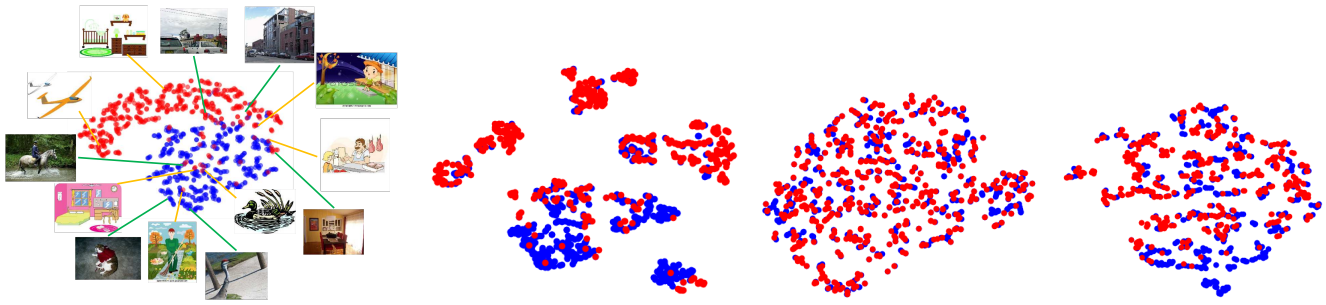
Implementation Details. In all experiments, we set the shorter side of the image to 600 following the implementation of Faster RCNN [29] with ROI-alignment [12]. We first trained the networks with learning rate 0.001 for 50K iterations, then with learning rate 0.0001 for 20K more iterations and reported the final performance. All models are trained with this scheduling and we reported the performance trained after 70K iterations. Without specific notation, we set λ as 1.0 and γ as 5.0. We implemented all methods with Pytorch [27]. Please see our supplemental material for the detail of the network architecture.

We compared our method with three baselines: FRCNN model, FRCNN with a baseline domain classifier, and domain adaptive FRCNN (DA-Faster) [5]. FRCNN model was trained only on source examples without any adaptation. The FRCNN with a baseline domain classifier has exactly the same architecture as our proposed weak-global alignment model, but its domain classifier is trained with cross-entropy loss in Eq. 5 and 6. The model does not have a local-level domain classifier. By comparing with this model, we can directly observe the effectiveness of our proposed weak alignment approach. Hereafter, we call the baseline *BDC-Faster*. *DA-Faster* [5] employs two domain classifiers, an image-level one for high-level features and an instance-level one for features cropped by the region proposal network. Both domain classifiers are trained by cross-entropy loss. In addition, it utilizes a technique called consensus regularization, which makes the outputs of two domain classifiers similar. Since we did not observe any benefit of the technique, we report the results without it. Since we implemented the method ourselve, the results reported in the original paper and in our paper are different. We denote their reported performance as *DA-Faster**.

4.1. Adaptation between dissimilar domains

We first show experiments on dissimilar domains, specifically, adaptation from real images to artistic images. We utilized the Pascal VOC Dataset as the real source domain [8]. This dataset contains 20 classes of images and their bounding box annotations. Following a prevalent evaluation protocol, we employed PASCAL VOC 2007 and 2012 training and validation splits for training, resulting in about 15k images. The target domain consists of either the Clipart or the Watercolor datasets [15]. Clipart contains comical images whereas Watercolor has artistic images. Clipart contains 1K images in total, which have the same 20 categories as PASCAL VOC. All images were used for both training (without labels) and testing. Watercolor contains 6 categories in common with PASCAL and 2K images in total. 1K training images were utilized during training and our model is evaluated on 1K test images. In this experiment, we used the ResNet101 [13] pre-trained on [7] as a backbone network. For other details see our supplemental material.

Results on Clipart. As shown in Table 1, our proposed method outperformed all baselines. Just by replacing the domain classifier's objective with the focal loss, MAP improved by 10.8% (25.6 to 36.4). In addition, the context vector based regularization and local alignment (C, L in



(a) Proposed (MAP: 36.4) (b) Baseline DC (MAP: 25.6) (c) Proposed (MAP: 29.1) (d) baseline DC (MAP: 27.6)

Figure 4. Visualization of features obtained by two different models. Blue: source examples, Red: target examples. Fig. (a) and (b) are the results of adaptation between dissimilar domains (from pascal to clipart). For Fig. (a), images with green lines are from PASCAL VOC (source). Images with orange lines are from Clipart (target). Our method does not match feature distributions strictly whereas the baseline method matches. However, our method outperformed the baseline with a large margin, which demonstrates the effectiveness of global-weak alignment. Fig. (d) and (c) are adaptation between similar domains (from Cityscape to FoggyCityscape). When the domains are very similar, the baseline method works well though our method performs better.

Table), further improved MAP. The performance on the source domain, PASCAL VOC, is shown in Table 2. Compared with the performance of the source only model, BDC-Faster and DA-Fastster significantly decrease its performance. This fact indicates that strictly aligning feature distributions between different domains can disturb the training for object detection while our method does not degrade the performance on the source domain.

We further visualized the features obtained by two models, our proposed global-level adaptation model and BDC-Faster in Fig. 4(a) and 4(b). The target features obtained by a baseline domain classifier are matched compactly with the source domain (Fig. 4(b)). On the other hand, with our proposed method (Fig. 4(a)), some features are aligned with the source features, but most of them are separated from source features. Source images usually focus on one or two objects whereas target images usually contain multiple images. Some target images focusing on single object are likely to be aligned with source as shown in the figure. Many existing methods for image classification aimed to match the feature distributions closely. However, this visualization implies that such distribution matching does not always help domain adaptive object detection.

Results on Watercolor. According to Table 3, our method outperformed the baseline methods. There was a large improvement on this domain. The improvement by the local alignment is especially large, about 3%, because the target images have a characteristic "painting" style. Therefore, the reducing the domain-gap based on local-level features improves the performance.

4.2. Adaptation between similar domains

In this experiment, we aim to analyze our method by evaluating the adaptation between very similar domains. We used Cityscape [6] as the source domain. The images

Table 3. AP on adpatation from PASCAL VOC to WaterColor (%). The definition of G, I, CTX, L is following Table 1.

Method	G	I	CTX	L	AP on a target domain bike	bird	car	cat	dog	prsn	MAP
Source Only					68.8	46.8	37.2	32.7	21.3	60.7	44.6
BDC-Faster	✓				68.6	48.3	47.2	26.5	21.7	60.5	45.5
DA-Faster	✓	✓			75.2	40.6	**48.0**	31.5	20.6	60.0	46.0
Proposed	✓				66.4	53.7	43.8	**37.9**	31.9	65.3	49.8
	✓		✓		71.3	52.0	46.6	36.2	29.2	**67.3**	50.4
	✓		✓	✓	**82.3**	**55.9**	46.5	32.7	**35.5**	66.7	**53.3**

Table 4. AP on adaptation from Cityscape to FoggyCityscape (%). The performance of our method is very near to oracle, which is trained on labeled target images.

Method	G	I	CTX	L	AP on a target domain bus	bcycle	car	bike	prsn	rider	train	truck	MAP
Faster RCNN					22.3	26.5	34.3	15.3	24.1	33.1	3.0	4.1	20.3
BDC-Faster	✓				29.2	28.9	42.4	22.6	26.4	37.2	12.3	21.2	27.5
DA-Faster	✓	✓			33.1	23.3	25.5	15.6	23.4	29.0	10.9	19.6	22.5
DA-Faster*	✓	✓			25.0	31.0	40.5	22.1	**35.3**	20.2	20.0	**27.1**	27.6
Proposed	✓				33.5	33.3	42.7	22.2	27.1	40.3	11.6	22.3	29.1
				✓	34.3	32.2	36.2	23.7	27.5	39.3	5.4	24.4	27.9
	✓		✓		**38.0**	31.2	41.8	20.7	26.6	37.6	19.7	20.5	29.5
	✓		✓	✓	36.2	**35.3**	**43.5**	**30.0**	29.9	**42.3**	**32.6**	24.5	**34.3**
Oracle					50.0	36.2	49.7	34.7	33.2	45.9	37.4	35.6	40.3

in the dataset are captured by a car-mounted video camera. As the target domain, we used FoggyCityscape datasets [34]. The images are rendered from Cityscape using depth information and it simulates the change of weather condition. The important difference from other adaptation scenario is that source and target images are originally the same one. Target images are generated from source images by adding fog noise. In such adaptation scenario, strictly aligning feature distributions should be effective because there exists a correct matching between source and target images. Both dataset have 2, 975 images in the training set,

Table 5. Results on adpatation from Sim10k to Cityscape Dataset (%). Average precision is evaluated on target images. FL ($\gamma = 3$)* indicates the experiments in which shorter side of image is scaled to 1000 during training and testing. *P* indicates pixel-level alignment, whether we used images generated by cyclegan during training. † indicates the performance when the context vector is zero-padded and not used for the output.

Method	G	I	CTX	L	P	AP on Car
Faster RCNN						34.6
BDC-Faster	✓					31.8
DA-Faster	✓	✓				34.2
DA-Faster*	✓	✓				38.9
Weak Align	✓	✓				35.8
Proposed (FL)	✓					36.4
	✓		✓			38.2 (38.3)†
	✓		✓	✓		40.1
	✓		✓		✓	41.5
	✓		✓	✓	✓	40.7
Proposed Method with different parameters						
EFL	✓		✓			38.7
FL ($\gamma = 3$)	✓		✓			42.3
FL ($\gamma = 3$)*	✓		✓		✓	47.7
Oracle						53.1

and 500 images in the validation set. We utilized the training set during training and evaluated on the validation set. Since Cityscapes dataset does not have bounding-box annotation, we take the tightest rectangles of its instance masks as groundtruth bounding boxes. We used the VGG16 model [37] as a backbone network following [5].

As shown in Table 4, our proposed method performed much better than the baseline methods. MAP of a model with only strong local alignment was 27.9. Combining strong local and weak global alignment boosted MAP to 34.3. The domain-shift is caused by fog noise, a local-level shift. Hence, strong local alignment largely contributed to the improvement. In this adaptation scenario, the method with a baseline domain classifier performs better than the source only model. This is because the target images have exactly the same layout and number/combination of objects. Thus, strong alignment between different domains was effective. The visualized features in Fig. 4 show completely different characteristics from the experiments on PASCAL to Clipart dataset. The features are matched in both methods. The results indicate that our proposed method performs both when two domains are dissimilar and similar.

4.3. Adaptation from synthetic to real images

We evaluate the performance of our model in an adaptation from synthetic images to real images. As the synthetic domain, we used Sim10k [16]. The dataset contains images of the synthetic driving scene, 10,000 training images which are collected from the computer game Grand Theft Auto (GTA). We employed the same architecture as used in the previous section. Following the protocol of [5], we evaluated detection performance on *car*. As a real domain, we used Cityscape. All training images are used during training for both domains. Average precision was evaluated on the validation split of the Cityscape. We set the value of $\lambda = 0.1$ following [5] in Eq. 12. We show the performance when varying the value of λ in our supplemental material. The two domains have similar layout in that both domains are driving scene images. However, the color and lighting are clearly different. In this respect, the two domains are more different than Cityscape and Foggycityscape are. We extensively evaluated our method by ablating some components. Moreover, we show the results using instance-level adaptation as proposed in [5]. We also show the comparison and results of combination with a model trained with images translated by CycleGAN [42]. We trained CycleGAN to translate different domains' images, then utilized the translated source images for training. Whether we employed the translated images is denoted by the colum of P in Table 5. The details are shown in supplemental material. In addition, we demonstrate that our idea of weak alignment can be achieved with a loss function other than focal loss. In Eq. 2, we set $f(p_t) = e^{-\eta p_t}$, which is a decreasing function with the value of p_t. We call the loss function exponential focal loss (EFL). We set $\eta = 5.0$.

The results are summarized in Table 5. Our method constantly performed better than the baseline models. Comparing the results of BDC-Faster (31.8) and our method with only global-level alignment (36.4), the weak feature distribution alignment outperformed the strict alignment. Setting the value of $\gamma = 3.0$ in Focal Loss significantly improved the performance. In addition, with regard to a model trained with EFL, we could observe the improvement over the baseline models. The results demonstrate that our idea of weak global alignment is effective and can be achieved by functions other than Focal Loss.

Context vector based regularization and local-level alignment further improved the performance. The performance did not degrade when we did not use the context vector in test phase as seen in the table. This implies that the network does not use the vector for the prediction whereas the performance improved compared to the model without the regularization. Therefore, the context vector seems to contribute to the regularization of the domain classifier.

We could not see a positive effect of instance-level adaptation (Weak Align in Table 5). Instance-level alignment utilizes the cropped features by region proposal networks, but the proposals may not localize objects in the target domain well, so it can hurt the performance of the model.

4.4. Analysis

Examples of detection results. We show the examples of detection results in Fig. 5. Even when the style of the images is different between the source and target, our

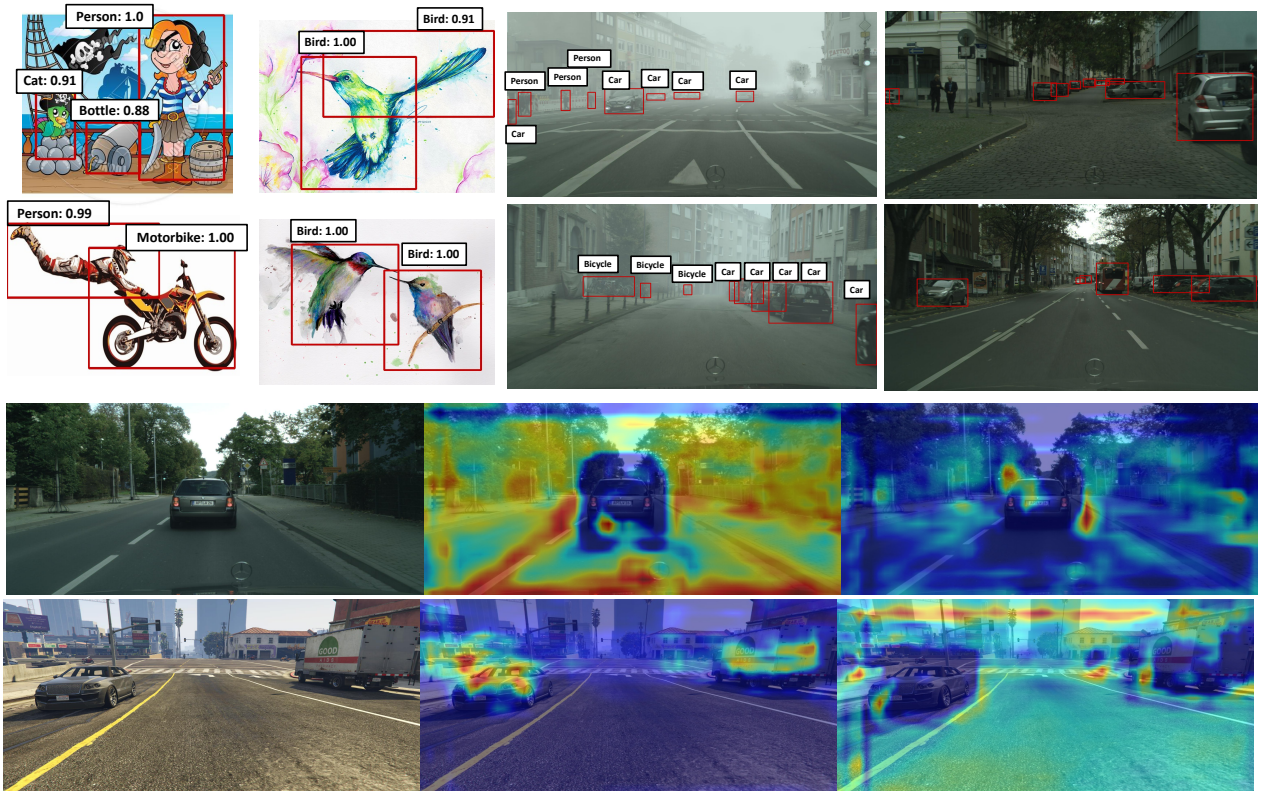


Figure 5. Upper: Examples of detection results on the target domain. From left to right column, Clipart, Watercolor, FoggyCityscape and Cityscape dataset. Bottom: Visualization of domain evidence using Grad-Cam. The evidence is obtained by the global-domain classifier. The pictures show results on target (Top) and source images (Bottom). From left to right, input images, images of evidence for the target, evidence of the source domain. The feature extractor seems to focus on deceiving the domain classifier in regions with cars.

model localizes objects correctly in these cases. As seen in Clipart's example, when the appearance of the objects is largely different, the detection results are not successful. Also, as seen in case of Watercolor, the detector tends to output multiple predictions to one object. In case of FoggyCityscape's examples, our model tends to assign one bounding box to multiple neighboring bicycles.

Visualization of domain evidence. To analyze the behavior of the feature extractor and domain classifier, we visualize the evidence for the global-level domain classifier's prediction using Grad-cam [36] in Fig. 5. We use Grad-cam to show the evidence (heatmap) for why the domain classifier thinks the image comes from the source or the target, for the adaptation from Sim10k to Cityscapes. Please see our supplemental material for other examples. For the target images, the domain classifier does not look at cars as the evidence for the target. Similarly, for source images, it also does not look at cars as the evidence for the source. This indicates that the feature extractor seems to focus on cars to deceive the domain classifier, which means that the feature extractor learns to partially align global-image features, specifically around cars.

5. Conclusion

In this work, we propose a novel approach for detector adaptation based on strong local alignment and weak global alignment for unsupervised adaptation of object detectors. Our key contribution is the weak alignment model, which focuses the adversarial alignment loss on images that are globally similar and puts less emphasis on aligning images that are globally dissimilar. Additionally, we design the strong domain alignment model to only look at local receptive fields of the feature map. Our method outperformed other existing methods with a large-margin in several datasets. Through extensive experiments, we verified the effectiveness of weak global and strong local alignment.

6. Acknowledgements

This work was supported by Honda, DARPA and NSF Award No. 1535797 and partially supported by JST CREST Grant Number JPMJCR1403, Japan.

References

[1] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan. A theory of learning from different domains. *Machine learning*, 79(1-2):151–175, 2010. 1, 2

[2] S. Ben-David, J. Blitzer, K. Crammer, F. Pereira, et al. Analysis of representations for domain adaptation. In *NIPS*, 2007. 1, 2

[3] H. Bilen and A. Vedaldi. Weakly supervised deep detection networks. In *CVPR*, 2016. 2

[4] P. P. Busto and J. Gall. Open set domain adaptation. In *ICCV*, 2017. 2

[5] Y. Chen, W. Li, C. Sakaridis, D. Dai, and L. Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In *CVPR*, 2018. 2, 3, 5, 7

[6] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 5, 6

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*. 2, 5

[8] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010. 2, 5

[9] Y. Ganin and V. Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, 2014. 2, 5

[10] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-adversarial training of neural networks. *JMLR*, 17(59):1–35, 2016. 1

[11] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*, 2014. 2

[12] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *ICCV*, 2017. 5

[13] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5

[14] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. A. Efros, and T. Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *ICML*, 2018. 1

[15] N. Inoue, R. Furuta, T. Yamasaki, and K. Aizawa. Cross-domain weakly-supervised object detection through progressive domain adaptation, 2018. 5

[16] M. Johnson-Roberson, C. Barto, R. Mehta, S. N. Sridhar, K. Rosaen, and R. Vasudevan. Driving in the matrix: Can virtual worlds replace human-generated annotations for real world tasks? *arXiv preprint arXiv:1610.01983*, 2016. 5, 7

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 1

[18] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 3

[19] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 2

[20] M.-Y. Liu, T. Breuel, and J. Kautz. Unsupervised image-to-image translation networks. In *NIPS*, 2017. 1

[21] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *ECCV*, 2016. 2

[22] M. Long, Y. Cao, J. Wang, and M. I. Jordan. Learning transferable features with deep adaptation networks. In *ICML*, 2015. 2

[23] M. Long, Z. Cao, J. Wang, and M. I. Jordan. Conditional adversarial domain adaptation. In *NIPS*, 2018. 2

[24] M. Long, H. Zhu, J. Wang, and M. I. Jordan. Unsupervised domain adaptation with residual transfer networks. In *NIPS*, 2016. 1, 2

[25] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley. Least squares generative adversarial networks. In *ICCV*, 2017. 4

[26] A. Odena, C. Olah, and J. Shlens. Conditional image synthesis with auxiliary classifier gans, 2017. 3

[27] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017. 5

[28] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, 2016. 2

[29] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 2, 5

[30] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *ECCV*, 2010. 1, 2

[31] K. Saito, Y. Ushiku, T. Harada, and K. Saenko. Adversarial dropout regularization. In *ICLR*, 2018. 1

[32] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *CVPR*, 2018. 2

[33] K. Saito, S. Yamamoto, Y. Ushiku, and T. Harada. Open set domain adaptation by backpropagation. In *ECCV*, 2018. 2

[34] C. Sakaridis, D. Dai, and L. Van Gool. Semantic foggy scene understanding with synthetic data. *IJCV*, 2018. 5, 6

[35] S. Sankaranarayanan, Y. Balaji, A. Jain, S. N. Lim, and R. Chellappa. Learning from synthetic data: Addressing domain shift for semantic segmentation. In *CVPR*, 2018. 2, 3, 4

[36] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, et al. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, 2017. 8

[37] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 7

[38] Y. Tang, J. Wang, B. Gao, E. Dellandréa, R. Gaizauskas, and L. Chen. Large scale semi-supervised object detection using visual and semantic knowledge transfer. In *CVPR*, 2016. 2

[39] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *CVPR*, 2017. 1, 2

[40] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell. Deep domain confusion: Maximizing for domain invariance. *arXiv:1412.3474*, 2014. 1, 2

[41] J. Zhang, Z. Ding, W. Li, and P. Ogunbona. Importance weighted adversarial nets for partial domain adaptation. In *CVPR*, 2018. 2

[42] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *ICCV*, 2017. 4, 7

[43] Y. Zou, Z. Yu, B. V. Kumar, and J. Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *ECCV*, 2018. 2