Adaptive Object Detection with Dual Multi-Label Prediction

Zhen Zhao, Yuhong Guo, Haifeng Shen, Jieping Ye

Problem: Cross-Domain Object Detection

- Problem Setting
  - A source domain with a large number of annotated images
  - A target domain with only unannotated images
  - Aim to learn a good object detector for the target domain

- Previous cross-domain object detection methods apply adversarial feature alignment to bridge domain divergence.

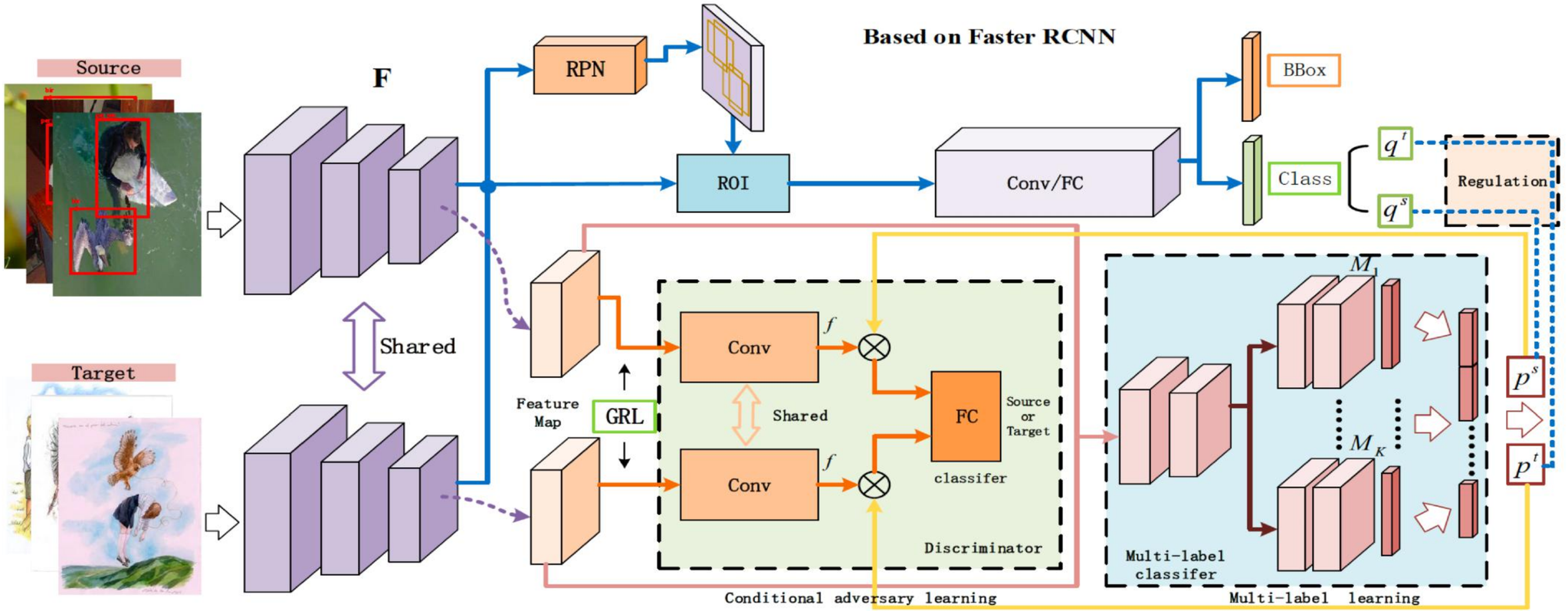
- However, an image can contain **multiple different objects** and the global image features can have **complex multimodal structures**. Standard image-level feature alignment can fail to handle this issue.

Proposed Approach: MCAR

- **MCAR:** Multi-label Conditional distribution Alignment and detection Regularization model

- The first work that exploits multi-label prediction as an auxiliary task for cross-domain object detection

MCAR: Overall Architecture



Auxiliary Multi-Label Prediction

- **Multi-label Conditional adversarial feature alignment (MC)**
- **Prediction consistency Regularization (PR)**

Multi-Label Prediction

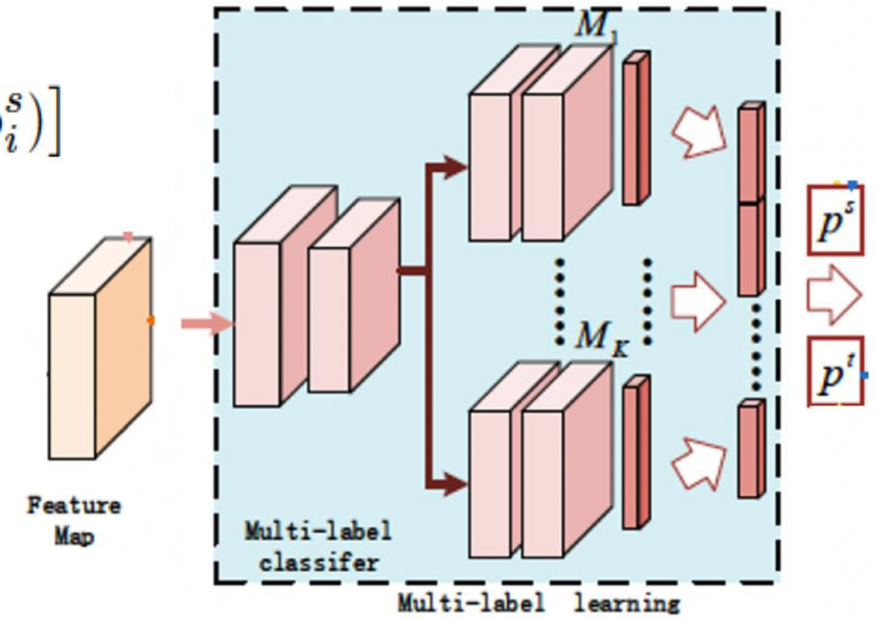
- Transform object detection into an object recognition problem:

  Bounding boxes' labels $c_i^s \longrightarrow y_i^s, \quad y_i^s \in \{0,1\}^K$

- Solve the object recognition task as a multi-label prediction problem
  - Train K binary classifiers by minimizing the following cross-entropy loss:

$$\mathcal{L}_{multi} = -\frac{1}{n_s}\sum_{i=1}^{n_s}\left[\mathbf{y}_i^{s\top}\log(\mathbf{p}_i^s) + (1-\mathbf{y}_i^s)^\top \log(1-\mathbf{p}_i^s)\right]$$

$$\mathbf{p}_{ik}^s = M_k(F(x_i^s))$$



Multi-Label Conditional Adversarial Feature Alignment

- **Use multi-label prediction results as the category information for the conditional adversary**, i.e., domain discriminator:

$$D(F(x_i), \mathbf{p}_i) = FC(f(F(x_i)) \otimes \mathbf{p}_i)$$

- Deploy a conditional adversarial training loss, L_{adv} , to perform multimodal feature distribution alignment

$$\min_F \max_D \quad \mathcal{L}_{adv} = -\frac{1}{2}(\mathcal{L}_{adv}^s + \mathcal{L}_{adv}^t)$$

$$\mathcal{L}_{adv}^s = -\frac{1}{n_s}\sum_{i=1}^{n_s}(1-D(F(x_i^s), \mathbf{p}_i^s))^\gamma \log(D(F(x_i^s), \mathbf{p}_i^s))$$

$$\mathcal{L}_{adv}^t = -\frac{1}{n_t}\sum_{i=1}^{n_t} D(F(x_i^t), \mathbf{p}_i^t)^\gamma \log(1-D(F(x_i^t), \mathbf{p}_i^t))$$

  - maintain the discriminability of the features

Prediction Consistency Regularization

- **Use multi-label prediction results to help object detection**
  - ***Obtain object level labels from the object detector*** by combing the prediction results from the total N region proposals : $Q \in [0,1]^{K \times N}$
  - Compute an overall **multi-object prediction probability vector *q*** by taking the row-wise maximum over Q: $q_k = \max(Q(k,:))$, and renormalize
  - **Enforce prediction consistency between p and q** by minimizing the KL divergence based loss:

$$\mathcal{L}_{kl} = \mathcal{L}_{kl}^{s} + \mathcal{L}_{kl}^{t}$$

$$\mathcal{L}_{kl}^{s} = \frac{1}{2n_s} \sum_{i=1}^{n_s} (KL(\mathbf{p}_i^s, \mathbf{q}_i^s) + KL(\mathbf{q}_i^s, \mathbf{p}_i^s))$$

$$\mathcal{L}_{kl}^{t} = \frac{1}{2n_t} \sum_{i=1}^{n_t} (KL(\mathbf{p}_i^t, \mathbf{q}_i^t) + KL(\mathbf{q}_i^t, \mathbf{p}_i^t))$$

Over End-to-End Learning

➢ End-to-End Learning

- The detection loss of the base Faster-RCNN model, denoted as L_{det}
- Combine the **detection loss**, the **multi-label prediction loss**, the **conditional adversarial feature alignment loss**, and the **prediction consistency regularization loss** together for end-to-end deep learning:

$$\begin{cases} \mathcal{L}_{all} = \mathcal{L}_{\text{det}} + \lambda \mathcal{L}_{adv} + \mu \mathcal{L}_{multi} + \varepsilon \mathcal{L}_{kl} \\ \\ \min\limits_{F} \max\limits_{D} \quad \mathcal{L}_{all} \end{cases}$$

where λ, μ, and ε are trade-o parameters that balance the multiple loss terms.

Experiments

Multiple cross-domain multi-object detection tasks under different adaptation scenarios:

- Domain adaptation from real to virtual images
  PASCAL VOC to **Watercolor2K** and **Comic2K** respectively

- Domain adaption from normal/clear images to foggy images
  Cityscapes to **Foggy Cityscapes**

Adaptation from Real to Virtual Scenes

- PASCAL VOC to Watercolor

Method	MC	PR	bike	bird	car	cat	dog	person	mAP
Source-only			68.8	46.8	37.2	32.7	21.3	60.7	44.6
BDC-Faster [31]			68.6	48.3	47.2	26.5	21.7	60.5	45.5
DA-Faster [2]			75.2	40.6	48.0	31.5	20.6	60.0	46.0
SW-DA [31]			82.3	**55.9**	46.5	32.7	35.5	66.7	53.3
SCL [34]			82.2	55.1	**51.8**	39.6	**38.4**	64.0	55.2
MCAR (Ours)	✓		**92.5**	52.2	43.9	**46.5**	28.8	62.5	54.4
	✓	✓	87.9	52.1	**51.8**	41.6	33.8	**68.8**	**56.0**
Train-on-Target			83.6	59.4	50.7	43.7	39.5	74.5	58.6

- PASCAL VOC to Comic

Method	MC	PR	bike	bird	car	cat	dog	person	mAP
Source-only			32.5	12.0	21.1	10.4	12.4	29.9	19.7
DA-Faster			31.1	10.3	15.5	12.4	19.3	39.0	21.2
SW-DA			36.4	21.8	29.8	15.1	23.5	49.6	29.4
MCAR (Ours)	✓		40.9	**22.5**	30.3	**23.7**	**24.7**	**53.6**	32.6
	✓	✓	**47.9**	20.5	**37.4**	20.6	24.5	50.2	**33.5**

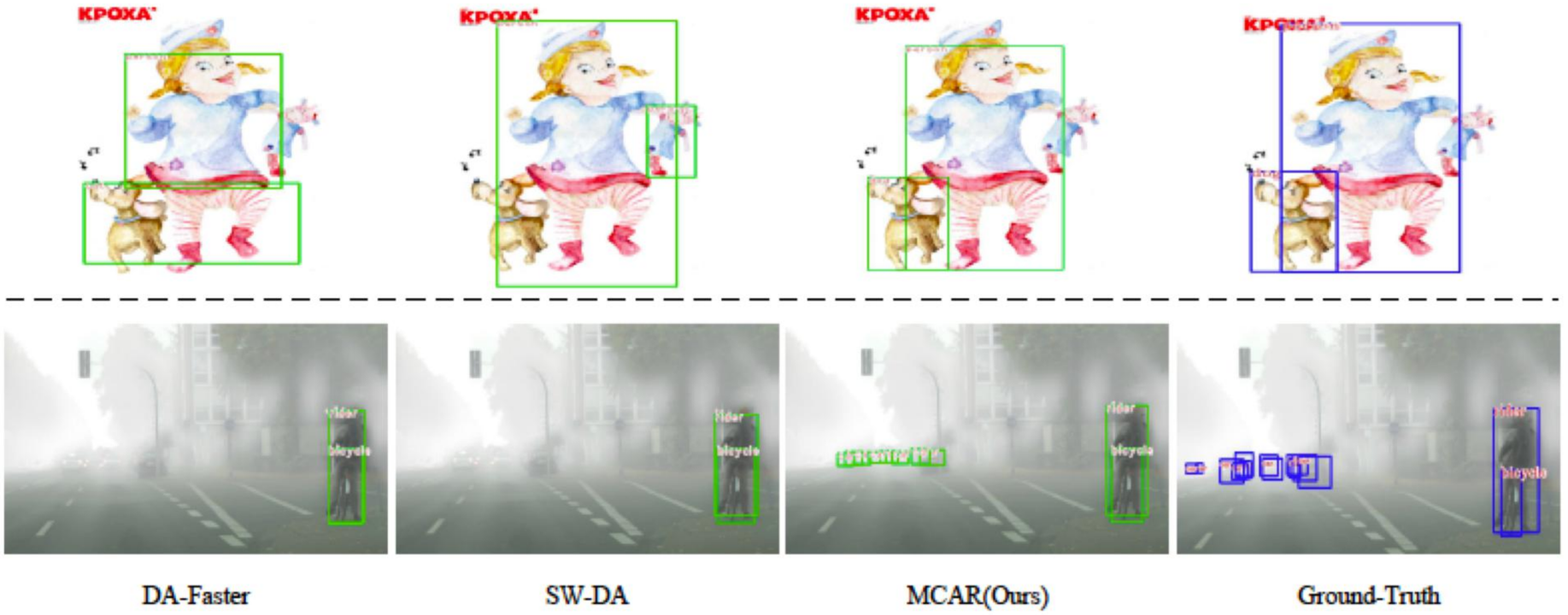
Adaptation from Normal to Foggy Scenes

- Cityscapes to Foggy Cityscapes

Method	MC	PR	person	rider	car	truck	bus	train	motorbike	bicycle	mAP
Source-only			25.1	32.7	31.0	12.5	23.9	9.1	23.7	29.1	23.4
BDC-Faster [31]			26.4	37.2	42.4	21.2	29.2	12.3	22.6	28.9	27.5
DA-Faster [2]			25.0	31.0	40.5	22.1	35.3	20.2	20.0	27.1	27.6
SC-DA [44]			33.5	38.0	**48.5**	26.5	39.0	23.3	28.0	33.6	33.8
MAF [17]			28.2	39.5	43.9	23.8	39.9	33.3	29.2	33.9	34.0
SW-DA [31]			**36.2**	35.3	43.5	30.0	29.9	42.3	32.6	24.5	34.3
DD-MRL [19]			30.8	40.5	44.3	27.2	38.4	34.5	28.4	32.2	34.6
MTOR [1]			30.6	41.4	44.0	21.9	38.6	40.6	28.3	35.6	35.1
Dense-DA [40]			33.2	**44.2**	44.8	28.2	41.8	28.7	30.5	36.5	36.0
SCL [34]			31.6	44.0	44.8	30.4	41.8	40.7	33.6	36.2	37.9
MCAR (Ours)	✓		31.2	42.5	43.8	**32.3**	41.1	33.0	32.4	36.5	36.6
	✓	✓	32.0	42.1	43.9	31.3	**44.1**	**43.4**	**37.4**	**36.6**	**38.8**
Train-on-Target			50.0	36.2	49.7	34.7	33.2	45.9	37.4	35.6	40.3

Qualitative Results

- PASCAL VOC to Watercolor
- Cityscapes to Foggy Cityscapes.



Thank You!