Domain Attention Model for Domain Generalization in Object Detection

Weixiong He[1,2,3], Huicheng Zheng[1,2,3(✉)], and Jianhuang Lai[1,2,3]

[1] School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China
[2] Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, Guangzhou, China
[3] Guangdong Key Laboratory of Information Security Technology, Guangzhou, China
hewx5@mail2.sysu.edu.cn,{zhenghch,stsljh}@mail.sysu.edu.cn

Abstract. Domain generalization methods in object detection aim to learn a domain-invariant detector for different domains. However, it is difficult to obtain a domain-invariant detector when there is large discrepancy between different domains. Based on the idea of biasing the allocation of available processing resources towards the most informative components of an input, attention models have shown promising performance on different tasks. In this paper, we provide a framework for addressing the issue of visual domain generalization with domain attention. Specifically, we build a domain attention block utilizing the source domain discrepancy to learn different weights for different source domains on the input features, so that the input features similar to the source domains will be enhanced and the features different from all the source domains will be suppressed. Thus we can obtain a domain-general representation effective for localization and classification in the proposed model. In order to demonstrate the merits of the proposed approach, we put forward a HD-16 dataset for object detection in different scenes. Extensive experiments on HD-16 dataset verify the effectiveness of the proposed approach.

Keywords: Domain generalization · Object detection
Attention model

1 Introduction

Object detection, the task that locates specific objects in images, is a fundamental problem in computer vision. In recent years, driven by the development of deep convolutional neural networks (CNNs) [14], many CNN-based object detection approaches [15,17,23,25] have been proposed and the performance of object detection was improved drastically. However, detectors trained on benchmark datasets would not always obtain satisfactory detection results when being applied to a new scene in the wild, due to the domain shift between the training source domains and the unknown testing target domains. In order to overcome the impact of domain shift, domain adaption (DA) methods [3,18,27] and

J.-H. Lai et al. (Eds.): PRCV 2018, LNCS 11259, pp. 27–39, 2018.
https://doi.org/10.1007/978-3-030-03341-5_3

domain generalization (DG) methods [1,3,6,7,20–22] are proposed to improve the performance in target domains. DA methods require target data to train a new model when facing a new target scene, and thus their performances depend largely on the distributions of target domains. Moreover, DA methods are based on the assumption that the target samples can be commodiously obtained, which is impractical in some cases. On the other hand, DG methods learn domain-invariant models without target samples, and can be more conveniently implemented in practice. The basic idea of DG methods is to combine source data in a way to produce models invariant for the specific target data, so that the model has satisfactory performance on different target scenes. However, existing DG methods seem to become degraded when the discrepancy between source domains and target domains is large, since the models trained on the source domains may not represent samples from the target scene well.

Based on the idea of biasing the allocation of available processing resources of an input, attention model [11,12,19,29] can dynamically weight the information of a signal. Therefore attention model can increase the ability to represent samples and has shown promising performances on different tasks. Nonetheless, little development is obtained in using the existing attention methods for domain generalization, because the labeled target samples are unobtainable and no supervised information is provided for biasing the suitable allocation of target domains.

In this work, we introduce a domain generalization approach for objection detection. Different from the previous work that tried to learn a domain-invariant model, we propose to utilize the discrepancy between different source domains to build an attention model and let the model put attention on the features that are similar to the source domains.

Our motivation comes from the observation that though source domains have different forms of distribution (Fig. 1(a)–(c)) with target domain (Fig. 1(d)), in which some of them have high similarity with the distribution of target domain while the others do not. If we treat these source domains all in the same way, the final distribution (Fig. 1(e)) may have a large gap with the target representation. However, a satisfactory result (Fig. 1(f)) can be obtained by combining these domains with different weights. Equivalently, target domain can also be resolved into sources domains after applying different weights on its domain specific features, and the output will be represented by the model easier. In order to achieve this goal, we propose the domain attention block to extract the domain specific weights of input and then differently weight each channel of the input, finally output an adaptive representation which is generalized for the model trained on the source domains. A large-scale human detection dataset with more than 90k images in 16 scenes is proposed to demonstrate the merits of proposed approach. The main contributions of this paper can be summarized as follows:

1. To address the domain generalization problem in object detection, we propose a novel domain attention model by introducing the domain attention blocks

to the baseline one-step detection model, which differently weight channels of the input according to the domain specific weights.
2. Given the images without domain labels in practice, we further present our method using the effective clustering method to generate pseudo domain labels.
3. We extensively perform comparative evaluations to show the superiority of our approach on the proposed dataset.

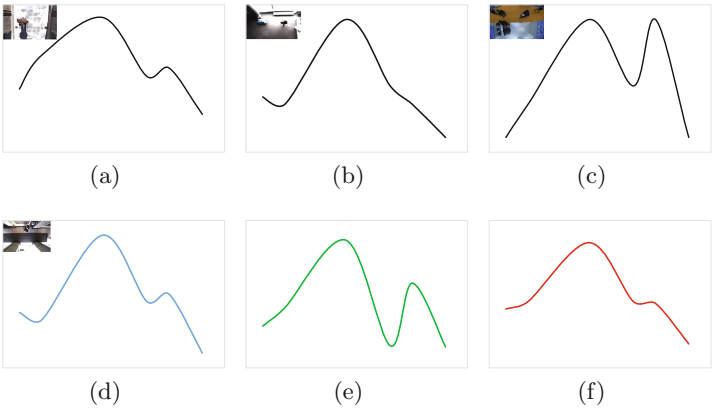


Fig. 1. Illustration of the distributions of different domains, (a)–(c) denotes the distributions of three source domains, (d) denotes the distribution of the target domain. The results of combining source representations in equal/unequal weights are shown in (e)/(f) respectively. Images in left corner of (a)–(d) come from different domains.

2 Related Work

2.1 Object Detection

Object detection has been a classical problem in computer vision, resulting in a plentitude of approaches. Classical work [4,5] usually formulated object detection as a sliding-window-based classification problem. Following the rise of deep convolutional neural networks (CNNs) [14] in computer vision, the performance of object detection was improved drastically. Among the large number of CNN-based approaches [8,15–17,23,24], two-step detectors [8,15,24] have received significant attention due to their performances. This line of work starts from R-CNN [8], which extracts region proposals from the image and classifies each region of interest (ROI) independently. Besides, one-step detectors [16,17,23] were proposed and popularly used in recent years due to their superiority in terms of speed. One-step detectors begin with YOLO [23], which treats object detection as a regression problem that jointly predicts the locations and confidence based

on the output features from convolutional network backbone. Developed from YOLO, SSD [17] exploits features from multiple convolutional layers to achieve a multi-scale prediction for object localization and obtain a satisfactory detection performance. Thus, we use SSD as the baseline detection model, and further improve its generalization ability for object detection in new target domains.

2.2 Domain Generalization

In the previous works, domain generalization problem is mainly addressed in two ways. On the one hand, some methods aggregate the information from source domains to learn a domain-invariant representation [1,20,21]. Specifically, [21] learns a domain-invariant transformation by minimizing the distance between domains. [1] simply put all the training data from different domains together to learn a SVM classifier. On the other hand, there are some works exploiting all information from the source domains to train a classifier or regulate its weights [13,30]. Specifically, [13] weights the classifiers to work well on an unknown dataset, and [30] fuses the scores of classifiers for a test sample. However, those methods become degraded when the discrepancy between source scenes and the target scene is large. In this paper, we use the domain attention block to weight the input features according to the domain specific weights of the current input features. Actually, the proposed method is similar but inherently different from the first kind of methods. In our work, we try to resolve the target domain into source domains by applying different weights on domain specific features and finally output an adaptive representation which is generalized for the model trained on the source domains.

2.3 Attention Mechanism

Attention is a tool to bias the allocation of available processing resources towards the most informative components of an input signal [11,12,19,29]. In recent years, attention mechanisms have achieved great success in a range of tasks such as object localization, image classification and sequence-based models [2]. Specifically, [29] introduces a powerful trunk-and-mask attention mechanism using a hourglass model. [11] proposes SE block, which is a lightweight gating mechanism specialised to model channel-wise relationships in a computationally efficient manner and enhance the representational power of basic modules throughout the network. In this work, we propose domain attention block which is developed from SE block [11] to solve the domain generalization problem in object detection. However, the proposed domain attention block has a goal entirely different from SE block. While SE block try to model channel-wise relationships using the spatial information, domain attention block models the domain specific weights of the input features and differently weights the features using these weights. As a whole, the proposed domain attention block has better performance and adaptation in domain generalization problem.

3 Proposed Method

In this section, we firstly introduce the structure of domain attention block in Sect. 3.1. After that, a framework for the proposed domain attention model will be described in Sect. 3.2. We further propose a general method in Sect. 3.3 to deal with the situation that no domain label is available.

3.1 Domain Attention Block

Figure 2 shows the structure of the domain attention block, which consists of two branches, i.e., the domain specific branch and the domain aggregation branch. Taking as input the feature maps X, the domain specific branch can extract the domain specific scores of X, which are the confidence of X belonging to various source domains. Once the domain scores are obtained, domain aggregation branch will aggregate these scores to generate domain specific weights and finally output the weighted feature maps. In the following, we present more details about the domain attention block.

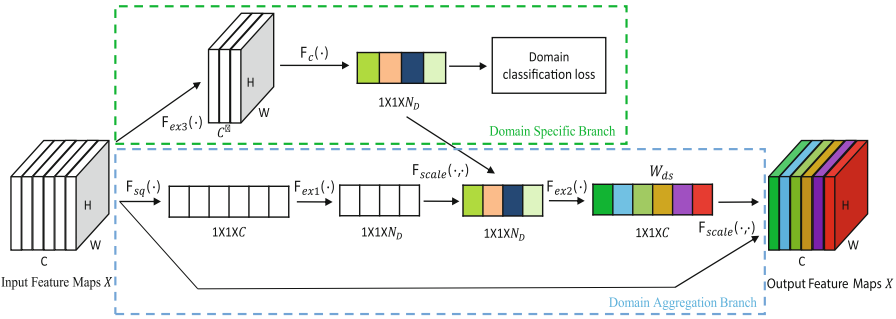


Fig. 2. Illustration of the structure of domain attention block.

Let $D = \{D_i\}_{i=1,2,...,N_D}$ denote the dataset consisting of N_D source domains, $i = 1, 2, ..., N_D$ is the domain labels of samples in D_i. Given the feature maps $X \in R^{H \times W \times C}$, we aim to learn a transformation $F(\cdot) : X \rightarrow \tilde{X}, \tilde{X} \in R^{H \times W \times C}$ which outputs the feature maps $\tilde{X}$ that are differently weighted for each channel according to the domain specific weights. We formulate $F(X)$ as

$$F(X) = F_{scale}(W_{ds}, X) \tag{1}$$

where W_{ds} represents the domain specific weights, and F_{scale} uses W_{ds} to differently weight each channel of X. Intuitively, a direct idea is using the domain scores as W_{ds} since we hope W_{ds} is specific for each domain. However, the number of domain scores is usually much less than the number of feature channels, thus it is unreliable to only take the domain scores as W_{ds} due to the imbalance between the scores and number of feature channels. What's more, the scores

can not directly be used as the domain specific weights because of the different dimensionality. Therefore we design function F_w which takes both the domain scores and X into account to obtain reasonable weights. Specifically we formulate the composite function F_w as

$$F_w = F_{scale}(F_{sq} \circ F_{ex1}, F_s) \circ F_{ex2} \tag{2}$$

where $\circ$ denotes an operation that composites two functions and F_{sq} generates channel-wise statistics using global average pooling. After the channel-wise statistics are generated, F_{ex1} processes them using a 1×1 convolution with N_D output channels, then F_{scale} weights these channels by the domain scores which are provided by a composite function F_s. Finally, F_{ex2} transforms these weighted features using 1×1 convolution with the same channel number as X, and applies a softmax operation to obtain the domain specific weights of X. When it comes to the generation of domain scores, we firstly extract the discriminative domain features of X, and hope that the domain score is easily generated based on these discriminative features. Therefore, we define F_s as

$$F_s = F_{ex3} \circ F_c \tag{3}$$

where F_{ex3} extracts the discriminative domain features of input X, then F_c transforms these discriminative domain into scores for each domain of the current input X. Specifically, given the input X, we firstly use 1×1 convolution as F_{ex3}, the output channel number of F_{ex3} is chosen as $C/16$ and C is the channel number of X. Then a fully connected layer with N_D outputs followed by the softmax operation is chosen as F_c. Actually, the above descriptions of F_s just build a structure for providing the score for each domain. The feasibility mainly relies on the accuracy of the domain scores output from F_c. Therefore, a softmax loss is added as the domain classifier loss on the output of F_c to maintain the accuracy of the domain scores.

3.2 The Overall Framework

We apply the domain attention block on the one-step detection model, resulting in the proposed domain attention model. An one-step detection model generally consists of two parts, the backbone convolution network and the unified classification/localization component. As shown in Fig. 3, we apply the domain attention block to the last several convolution layers in the backbone. There are two main reasons why we apply the attention block in such format. Firstly, discrepancy between different domains is more capturable in the high-level semantic information from top convolution layers [28]. Secondly, though difference between domains is existed, they still share some common information which are reflected in bottom layers [18]. As for the number of layers for applying domain attention block, we regard it as a hyper-parameter in the proposed method.

3.3 General Situation Without Domain Labels

We discuss our method above based on the assumption that the total dataset D consists of a certain number of source datasets, and there is a priori domain

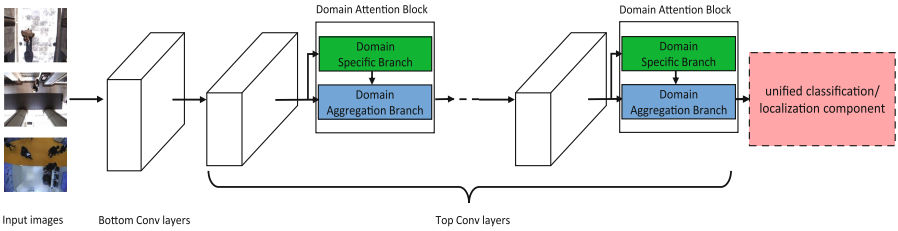


Fig. 3. Illustration of domain attention model.

label for each image, which means we know corresponding source dataset for each image coming from D. Actually, this assumption is not always satisfied because it is time consuming to identify the relationships between each image and its corresponding source dataset. In the general condition, the number of source datasets and the relationships between the images and the source datasets are unknown.

In this situation, D is represented as $\{D_i\}_{i=1,2,\ldots,N}$ where $N \geq 1$ and N is unknown. We adjust the proposed method by a simple preprocessing for the total dataset D. Specifically, we firstly assume the certain number n of source datasets, then we use unsupervised clustering methods to generate the source dataset and label each image with a pseudo domain label. After the preprocessing above, the dataset D will conform to the assumption of Sect. 3.1, and the following process is the same as Sect. 3.1. The experimental results prove that this preprocessing is effective when there is not much difference between the assumption n and the ground truth N.

4 Experiments

In this section, we evaluate the proposed domain attention model for domain generalization in object detection. We construct a human detection dataset with images from 16 scenes (HD-16) for evaluation. Extensive experiments are conducted in order to demonstrate the merits of the proposed method.

4.1 HD-16 Dataset

HD-16 is a large human detection dataset which has $93,371$ images in total captured by the top-view cameras in 16 different scenes. The number of images of each scene various from $1,362$ to $17,510$. Each image in the dataset is in the top-view and at the scale of 320×240 in pixel. HD-16 is challenging due to the large discrepancy in uncontrolled illumination and background between the images from different scenes. Figure 4 shows some examples of HD-16.

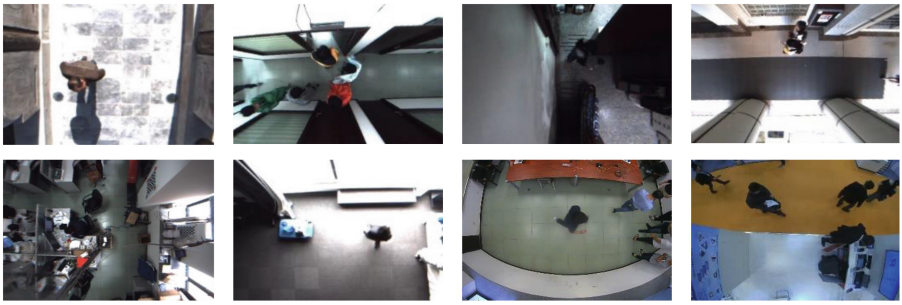


Fig. 4. Example images from different scenes in HD-16.

4.2 Experiment Setting

Baseline Detector. In experiment, we choose the SSD [17] as the baseline detector due to its outstanding performance in multi-scale object detection. In order to demonstrate the generalization of the proposed method, we respectively choose VGG-16 network [26] and MobileNet [10] as the backbone of SSD, and then apply the domain attention model on them for comparison.

Training Strategy. In the training process, we adopt the same data augmentation methods in SSD [17]. Moreover, we set the initial learning rate to 5×10^{-3} and the max-iteration of the training process is $300,000$.

Dataset Partition and Evaluation Protocol. Following the ordinary experimental protocol [18,28] for domain generalization datasets, we partition the HD-16 based on scenes. Specially, we randomly choose 12 scenes for training and the other 4 scenes for testing, resulting in a training set consisting of $65,534$ images and a testing set with $27,837$ images. For simplicity, we simply denote these 4 testing scenes as $T1 - T4$ and the combined testing set as C. Following the general criteria, we adopt mean average precision (mAP) with IoU of 0.5 for evaluation on HD-16.

4.3 Experiments on VGG-16 Based SSD

We firstly compare the proposed methods with SSD [17] based on VGG-16 network [26]. Because the image in the dataset is 320×240 in pixel and a person covers limited pixels in the image, we remove the conv7-9 layers as these layers have bigger receptive field than the actual area of a person. Then, we apply domain attention blocks after fc7, conv6_1, and conv6_2 in VGG-16. Since the proposed method develops from SENet [11], for fair comparison we further experiment on SSD with backbone of SENet (VGG-16 based), the results are shown in Table 1. It is evident from Table 1 that our method outperforms both competitors, for example, surpassing all compared methods by 3.7%(62.2%-56.0%),

Table 1. mAP(%) on VGG-16 based SSD.

	$T1$	$T2$	$T3$	$T4$	C
SSD [17]	56.0	57.0	71.5	72.8	56.4
SSD-SENet [11]	58.5	56.4	73.2	73.2	56.9
Ours	**62.2**	**59.3**	**75.9**	**74.7**	**59.8**

2.3%(59.3%-57.0%), 2.7%(75.9%-73.2%), 1.5%(74.7%-73.2%) and 2.9%(59.8%-56.9%) on $T1, T2, T3, T4$, and C, respectively. This indicates the advantages of the proposed method in handling domain generalization. The performance superiority is mainly because the proposed method effectively weights the input features and outputs adaptive representations which are generalized for the model trained on the source domains.

4.4 Experiments on MobileNet Based SSD

We evaluate the benefits of the proposed methods when integrate with other CNN architectures in addition to VGG-16. Specially, we select MobileNet architecture [10] for particularly testing the potentials in mobile vision application. For the same reason stated in Sect. 4.3, we remove conv14-17 and apply domain attention blocks after conv12 and conv13 in MobileNet. Table 2 shows the generic capability of the proposed method in weighting the input features and outputting adaptive representations for domain generalization when combined with a smaller MobileNet CNN architecture.

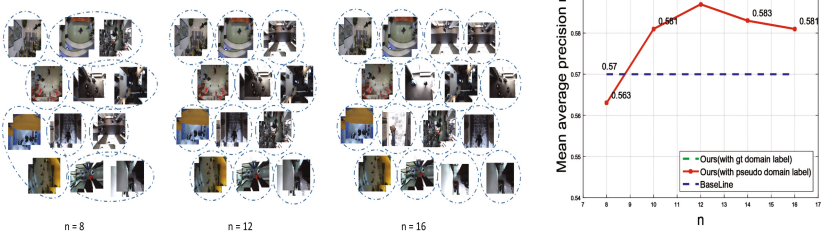
Table 2. mAP(%) on MobileNet based SSD.

	$T1$	$T2$	$T3$	$T4$	C
SSD [17]	55.1	53.3	65.7	72.0	55.9
SSD-SENet [11]	52.2	58.8	**67.2**	71.0	56.3
Ours	**56.2**	**59.6**	66.6	**72.3**	**58.9**

4.5 Experiments on General Situation Without Domain Labels

We further evaluate the proposed method in general situation without domain labels, which was discussed in Sect. 3.3. VGG-16 based SSD [17] is used for the baseline. Moreover, the result of the proposed method with ground-truth domain labels is used for comparison. For the training images without domain labels, we firstly choose n as the domain number. Specially, we choose the k-means algorithm [9] as the unlabeled cluster method by setting the hyper-parameter $k = n$. As for the representation of each image used in k-means algorithm, we simply resize each image to 40×30 and flatten to a feature vector. For simplicity, we use gt to denote the ground-truth domain number. We adjust n from 8 to 16,

Fig. 5(a) shows the clustering results and Fig. 5(b) shows the results on $T2$. We can infer from Fig. 5(a) that when $n < gt$, several domains tend to be merged into a cluster, and a domain is splited into several clusters when we set $n > gt$. It's the same as we expected that the proposed method has the best result when $n = gt = 12$, and the mAP is 0.6% lower (58.7% vs 59.3%) than the situation without domain labels. Moreover, the proposed method still outperforms the baseline (57.0%) in most situations, even when we set $n = 16$ that produces great deviation between the pre-set domain number and gt. In the situation that $n = 8$, the experiment result of the proposed method is worse than the baseline, we hold the opinion that when n is far less than gt, the model tends to treat all the samples in the same way and domain specific features are insufficient. Therefore, the domain attention block will degrade the performance to a small degree as the domain attention block may provide inaccurate weights.



(a) (b)

Fig. 5. (a) Clustering results when $n = 8, 12, 16$. (b) Evaluation of $T2$ on the general situation.

4.6 Model Complexity

Evaluation is also carried out on the proposed model from the aspect of model complexity. The model complexity of SSD based on VGG-16 and MobileNet will be used as the baseline for comparison. To further demonstrate the superiority of the proposed method in terms of model complexity, we also compare with the SENet-based SSD model, which has an attractive advantage in model complexity. We select the combined testing set C for evaluation. Table 3 shows the experiment result, the number in bracket denotes the percent of added capacity compared to the baseline. We can infer from Table 3 that SENet costs larger model capacity and obtains lower improvement (0.4%–0.5%) compared to the proposed method. Furthermore, the proposed method greatly improves the mAP (3%–3.4%) with less than 10% (1.9% in VGG-16) additional model capacity.

Table 3. Comprehensive evaluation on model complexity and detection performance

	Model capacity (MB)	mAP (%)
SSD-VGG16	88.0	56.4
SSD-VGG16 + SENet	90.0 (2.3%)	56.9
Ours-VGG16	89.7 (1.9%)	59.8
SSD-MobileNet	12.8	55.9
SSD-MobileNet + SENet	14.6 (12.8%)	56.3
Ours-MobileNet	14.0 (8.5%)	58.9

5 Conclusion

In this paper, we propose a domain attention model to solve the domain generalization problem for object detection in novel target domains. Based on the idea that target domain can be resolved into the sources domains, we propose to build a domain attention block by utilizing the discrepancy between different source domains, to weight the input data which contains domain specific features. The proposed approach is built on the state-of-the-art one-step object detector SSD and can be trained end-to-end using the standard SGD optimization. Moreover, we construct a HD-16 dataset for object detection in different scenes to demonstrate the merits of the proposed approach. Extensive experiments on HD-16 dataset have demonstrated the merits of the proposed approach.

Acknowledgement. This work was supported by National Natural Science Foundation of China (U1611461), Special Program for Applied Research on Super Computation of the NSFC-Guangdong Joint Fund (the second phase, No. U1501501), and Science and Technology Program of Guangzhou (No. 201803030029).

References

1. Blanchard, G., Lee, G., Scott, C.: Generalizing from several related classification tasks to a new unlabeled sample. In: NIPS, pp. 2178–2186 (2011)
2. Bluche, T.: Joint line segmentation and transcription for end-to-end handwritten paragraph recognition. In: NIPS, pp. 838–846 (2016)
3. Chen, Y., Li, W., Sakaridis, C., Dai, D., Van Gool, L.: Domain adaptive faster R-CNN for object detection in the wild. arXiv preprint arXiv:1803.03243 (2018)
4. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR, vol. 1, pp. 886–893. IEEE (2005)
5. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes (VOC) challenge. Int. J. Comput. Vis. **88**(2), 303–338 (2010)
6. Ghifary, M., Balduzzi, D., Kleijn, W.B., Zhang, M.: Scatter component analysis: a unified framework for domain adaptation and domain generalization. T-PAMI **39**(7), 1414–1430 (2017)

7. Ghifary, M., Bastiaan Kleijn, W., Zhang, M., Balduzzi, D.: Domain generalization for object recognition with multi-task autoencoders. In: ICCV, pp. 2551–2559 (2015)
8. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR, pp. 580–587 (2014)
9. Hartigan, J.A., Wong, M.A.: Algorithm as 136: a K-means clustering algorithm. J. Royal Stat. Soc. **28**(1), 100–108 (1979)
10. Howard, A.G., et al.: Mobilenets: efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861 (2017)
11. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. arXiv preprint arXiv:1709.01507 (2017)
12. Itti, L., Koch, C., Niebur, E.: A model of saliency-based visual attention for rapid scene analysis. T-PAMI **20**(11), 1254–1259 (1998)
13. Khosla, A., Zhou, T., Malisiewicz, T., Efros, A.A., Torralba, A.: Undoing the damage of dataset bias. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7572, pp. 158–171. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33718-5_12
14. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: NIPS, pp. 1097–1105 (2012)
15. Lin, T.Y., Dollar, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: CVPR, pp. 2117–2125. IEEE (2017)
16. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollar, P.: Focal loss for dense object detection. In: CVPR, pp. 2980–2988. IEEE (2017)
17. Liu, W., et al.: SSD: single shot multibox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2
18. Long, M., Cao, Y., Wang, J., Jordan, M.I.: Learning transferable features with deep adaptation networks. arXiv preprint arXiv:1502.02791 (2015)
19. Mnih, V., Heess, N., Graves, A., et al.: Recurrent models of visual attention. In: NIPS, pp. 2204–2212 (2014)
20. Motiian, S., Piccirilli, M., Adjeroh, D.A., Doretto, G.: Unified deep supervised domain adaptation and generalization. In: ICCV, pp. 5716–5726 (2017)
21. Muandet, K., Balduzzi, D., Schölkopf, B.: Domain generalization via invariant feature representation. In: ICML, pp. 10–18 (2013)
22. Niu, L., Li, W., Xu, D., Cai, J.: An exemplar-based multi-view domain generalization framework for visual recognition. T-PAMI **29**(2), 259–272 (2016)
23. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: CVPR, pp. 779–788. IEEE (2016)
24. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. T-PAMI **39**(6), 1137–1149 (2017)
25. Shen, Z., Liu, Z., Li, J., Jiang, Y.G., Chen, Y., Xue, X.: DSOD: learning deeply supervised object detectors from scratch. In: CVPR, pp. 1919–1927. IEEE (2017)
26. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
27. Tzeng, E., Hoffman, J., Darrell, T., Saenko, K.: Simultaneous deep transfer across domains and tasks. In: ICCV, pp. 4068–4076. IEEE (2015)

28. Tzeng, E., Hoffman, J., Zhang, N., Saenko, K., Darrell, T.: Deep domain confusion: Maximizing for domain invariance. arXiv preprint arXiv:1412.3474 (2014)
29. Wang, F., et al.: Residual attention network for image classification. arXiv:1704.06904 (2017)
30. Xu, Z., Li, W., Niu, L., Xu, D.: Exploiting low-rank structure from latent domains for domain generalization. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8691, pp. 628–643. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10578-9_41