Radar as a Teacher: Weakly Supervised Vehicle Detection using Radar Labels

Simon Chadwick and Paul Newman

***Abstract*— It has been demonstrated that the performance of an object detector degrades when it is used outside the domain of the data used to train it. However, obtaining training data for a new domain can be time consuming and expensive. In this work we demonstrate how a radar can be used to generate plentiful (but noisy) training data for image-based vehicle detection. We then show that the performance of a detector trained using the noisy labels can be considerably improved through a combination of noise-aware training techniques and relabelling of the training data using a second viewpoint. In our experiments, using our proposed process improves average precision by more than 17 percentage points when training from scratch and 10 percentage points when fine-tuning a pre-trained model.**

I. Introduction

Supervised learning methods have achieved ever-higher levels of performance in recent years. However, the requirement for large amounts of hand-labelled training data makes it labour intensive to deploy them in new domains. While much research effort has been devoted to improving the generalisation of models between domains, there is still a significant gap in performance if no labelled data from the new domain is available [1]. An alternative is to generate labels for a new domain using an automated method which takes advantage of circumstances that are not available at runtime (for example, additional sensors or running time backwards on recorded data). In this work we use a radar to automatically generate noisy labels for the task of detecting vehicles in images. We show how we can clean these labels to give good detector performance without the need for hand-labelling.

Radar is an appealing sensor to work with as it has a number of characteristics that are complementary to more commonly used sensors. For example, it is robust to most forms of environmental conditions (such as rain or fog). In this work we leverage its long range and the direct radial velocity measurements provided by the Doppler effect. We make the naïve assumption that radar targets with velocities that could not be caused by the motion of the data collection vehicle are generated by other vehicles moving nearby. This allows us to inexpensively generate copious amounts of training data without any hand-labelling effort.

Unlike LIDAR, radar provides velocity information without any need to associate between consecutive timestamps. This velocity information is the distinguishing characteristic that we use to identify what to label. As a result, labels are only created for moving vehicles. However, as the velocity information is not required at inference time (only images are used), the detector can be deployed on platforms without radar fitted and is free to generalise to stationary vehicles which would not be identified by the radar.

Authors are from the Oxford Robotics Institute, Dept. Engineering Science, University of Oxford, UK. {simonc,pnewman}@robots.ox.ac.uk

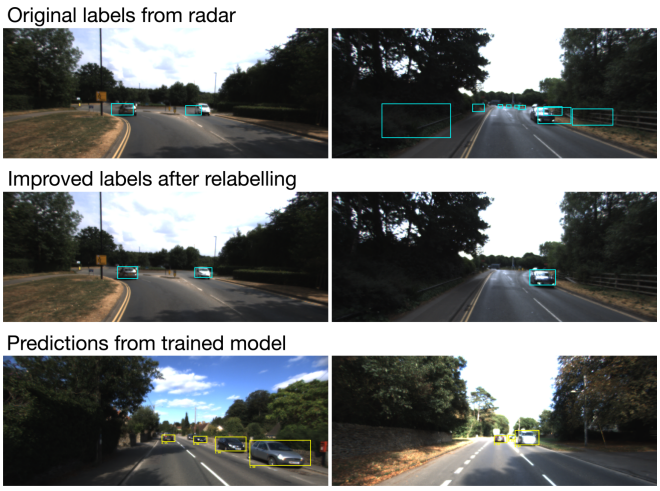

Fig. 1. Using only radar targets as labels we are able to train an effective vehicle detector using a noise-aware training process. We automatically generate the initial labels from radar data (*top*) then train a model using a modified version of the co-teaching process [2][3] to handle the noise. The training data is then relabelled using the first model, incorporating detections from a zoom image of the same scene to mitigate overfitting (*middle*). The new labels are used to train an improved model which provides effective vehicle detection from images alone (*bottom*).

As the labels we generate are noisy we use two processes to mitigate the effect of this noise. Firstly, we train our network using a modified version of the co-teaching process [2][3] which attempts to prevent erroneous labels from being used for parameter updates. Secondly, we use a relabelling process exploiting cameras of different focal lengths to both clean the labels and mitigate the degree of over-fitting that would occur if only a single image was used. An overview of the process is shown in Fig 1.

In a range of experiments presented below we show the benefit provided by each element of the process and demonstrate that good performance can be achieved despite the high level of noise in the labels.

II. Related Work

Supervised learning methods are the best performing machine learning techniques for many computer vision tasks. However, it is well known that the simplest way to achieve good performance is to increase the amount of available

training data. Given the cost of creating labelled datasets, there is a huge amount of research devoted to reducing manual labelling effort, thereby allowing bigger training sets without the additional cost. Some work focuses on "domain adaptation" - the practice of training a model using an existing labelled dataset in such a way that when the model is used in a new domain (for which no labels exist) any reduction in performance is minimised. While this problem is often studied in the context of semantic segmentation [4], some methods are applicable regardless of the task [5][6]. In [7] the particular case of domain adaptation for object detection is addressed by introducing losses to encourage the features generated to be agnostic to the domain of the image. Object detection is also targeted in [8] by incorporating elements of adversarial training, a common thread in much of the recent domain adaptation work. One particularly popular application is domain adaptation for models trained using synthetically generated data [9].

A second approach is to make do with data that is labelled in a cheaper fashion. In object detection this so-called weakly supervised task often takes the form of image level text labels. It was noted in [10] that methods for tackling this problem can fail by detecting only the most salient part of an object. This has led to efforts to counteract this failure mode [11][12][13]. The weakly supervised detection problem is tackled in [14] with the benefit of full labels in a neighbouring domain. This is exploited by using domain adaptation techniques to generate "pseudo labels" in the target domain. Many weakly supervised methods work from a set of object proposals (such as those generated by [15] or [16]) attempting to determine which proposal corresponds to the image-level label. The system of [17] aims to improve that process by using a model trained on motion cues to help rank the proposals. Regardless of the efficacy of these methods, some labelling effort is still required to generate the image-level labels.

When using alternative labelling techniques (for example, so-called "webly" supervised learning where labels are harvested from the internet) it is highly likely that the labels will contain some mistakes. This noise in the labels has a detrimental effect on the learning process, reducing the final performance of the trained detector. There has been a significant amount of effort devoted to methods that aim to reduce the effect of noisy labels. A number of these works focus on classification tasks such as [18]. In [19], Sukhbaatar *et al.* aim to learn the class noise transition matrix from the data by inserting an additional linear layer. Recently, [20] proposed a meta-learning objective to increase robustness to noisy labels. It also includes an iterative training process in which those examples for which the previous model gives a low score to the supposed ground-truth are excluded from the re-training.

A different approach to reducing labelling effort is to use unsupervised representation learning to train a model that generates a representation of the input from which the final task (in our case object detection) can be learned using a much smaller quantity of labelled data than would otherwise be the case. The current state-of-the-art in these approaches is [21] which learns a representation that maximises the mutual information between the latent space and future samples in a sequence — for images a sequence is generated by predicting patches below the current patch. An extension of this approach that may be applicable in multi-sensor settings (such as ours) is introduced in [22] where contrasts are made between different views rather than different elements of a sequence. Another method of learning representations is introduced in [23], where motion cues are used to segment moving objects as training data.

Our work is poised between the weakly supervised methods, where the labels are known to be incomplete, and methods related to training models from noisy labels.

Fig. 2. The good (*top*), the bad (*middle*) and the ugly (*bottom*) of using radar for labelling. We leverage a forward facing cruise control radar and use a fixed size prior to initially label those targets that do not match the platform's ego-motion. It can be seen that the labelling method, though crude, is remarkably accurate in many cases. However, it can also be seen that there is label noise, such as multiple bounding boxes for a single vehicle (*middle*) and ego-motion estimation errors leading to the inclusion of stationary targets (*bottom*). This noise necessitates the use of the training process that we propose.

III. Generating Labels from Radar

The radar that we use, a Delphi ESR 2.5 pulsed Doppler ADAS radar, does not provide access to the raw radar signal. Instead, it provides variable length lists of targets where each target consists of range, bearing, amplitude and range rate (radial velocity). With the assumption that each target occurs in the plane of the radar (an assumption that does not always hold, see Fig 4) this allows the $\mathbb{SE}3$ position of each target relative to the radar to be calculated.

Targets may represent returns from stationary or moving objects. As radar provides a clear velocity signal, we use that to distinguish moving vehicles from background (on the assumption that all moving objects are vehicles). We first subtract the ego-motion of the data collection vehicle, then take the subset of targets for which the resulting velocity is above a set threshold. Given the set of targets that correspond to moving vehicles these can then be approximately labelled

in image space by proposing a fixed size cuboid at the location of the radar target and projecting that cuboid into the image. Although crude, this yields plausible bounding boxes in a high proportion of cases (see Fig 2).

The obvious failure cases for this labelling method are stationary vehicles or vehicles whose primary motion is tangential rather than radial relative to the ego-vehicle (even if those cases where motion is entirely tangential are quite limited and brief). As a result of these failure cases and the assumptions made (for example the fixed size vehicle prior) the labels are sufficiently noisy that steps have to be taken to reduce the effect of this label noise.

IV. Co-teaching

To improve trained detector performance when learning from noisy labels, we make use of a modified version of the co-teaching framework [2]. Co-teaching aims to exclude gradients originating from examples that are mislabelled. It is based on the observation that simpler patterns are more easily learned during training [24] and that in these terms the objects that we aim to detect form a simpler pattern than any noise present in the labels. The implication of this is that, at intermediate points during the training, the losses of the two types of examples will diverge (i.e. clean label examples will be learned more quickly and so will have lower losses than examples with noisy labels).

To exploit this effect, co-teaching operates by training two identical but differently initialised networks in parallel. For each batch, each network informs the other which examples have the lowest loss and hence are the ones that should be used to provide gradients for parameter updates. The proportion of the batch used for updates reduces over time from the full batch at the start of training (when no learning has taken place so there is nothing to differentiate between examples) down to a subset that excludes a proportion equal to the estimated fraction of labels that are noisy.

The modifications introduced in [3] are shown to improve performance in an object detection setting. This is done by adapting the framework to exclude updates on an object-by-object basis (which is more suitable to object detection) rather than an image-by-image basis (as used for classification in the original paper), as shown in Figure 3. In addition to the modifications of [3], instead of setting a hyper-parameter to control the rate at which examples are excluded from training, we make use of the percentile moving average. Inspired by self-paced curriculum learning from [25] and [26], after a burn-in period we exclude losses above the percentile moving average (where the percentile is set according to the expected noise fraction). This also helps when batch sizes are small as the high/low loss distinction is then more consistent between each batch regardless of the composition of a specific batch.

V. Relabelling

Having trained an initial model in our domain using the labels derived from radar we then use that model to relabel our examples to give a cleaner label set (the process is shown in Fig. 4). Ordinarily, using a model to relabel its own training data would lead to considerable over-fitting. To reduce this effect we make use of a second camera with a zoom lens, capturing the scene concurrently with the wide-angle camera (note that only the wide-angle camera is required at test time). We run our trained model over the training set images of both cameras and combine the detections from both cameras. Using this method, the labels are refined using the more detailed zoom lens images. In addition, it also acts to perturb the new labels (away from the wide-angle only predictions) reducing overfitting (see results in Table I).

More specifically, we first train a model using the wide-angle images of the training set using the labels derived from radar. That model is then used to make predictions on every image of the training set from both the wide-angle images and the matching zoom lens images. The predictions from each wide-angle image and its matching zoom lens image are then thresholded based on the confidence of each prediction and combined. This combination is performed using the procedure described in [27]. In brief, the zoom lens predictions are reprojected into the wide-angle image by assuming the two cameras are coincident (in practice they are $\approx$ 30mm apart) which allows the image points to be transferred without knowledge of distance. The combined label set is formed by using the zoom lens predictions in the overlapping region subject to an intersection metric [27] at the boundary and the wide-angle labels elsewhere. In the case where co-teaching is used, two functional networks are trained simultaneously. Consequently, we optionally use both networks to make predictions on the zoom lens images. We then combine these two zoom lens sets by concatenating the predictions and using non-maxima suppression to remove highly overlapping predictions.

VI. Network Details

We use a network based on SSD [28], a single-stage object detection network. SSD operates by making predictions of class and bounding box offset for each one of a set of anchor boxes of different sizes and aspect ratios. These anchor boxes are repeated at each grid cell of a number of feature maps of different sizes. The dense predictions are then filtered based on class probability (the class prediction includes a "no object" option) and non-maxima suppression. We use a relatively small ResNet18 backbone [29] extended with an additional block that provides input to a larger scale feature map. In the case of single GPU training we are constrained in the size of network that can be used by the co-teaching procedure that requires two networks to be trained simultaneously. We use a batch size of 8 and the Adam optimiser [30] with an initial learning rate of $1e^{-4}$ reduced to $1e^{-5}$ after 55k iterations. We use L2 weight decay of $1e^{-3}$ as well as standard image augmentation processes including horizontal flips, random crops and colour perturbations.

As the number of labelled images decreases after relabelling we train for a fixed number of iterations (rather

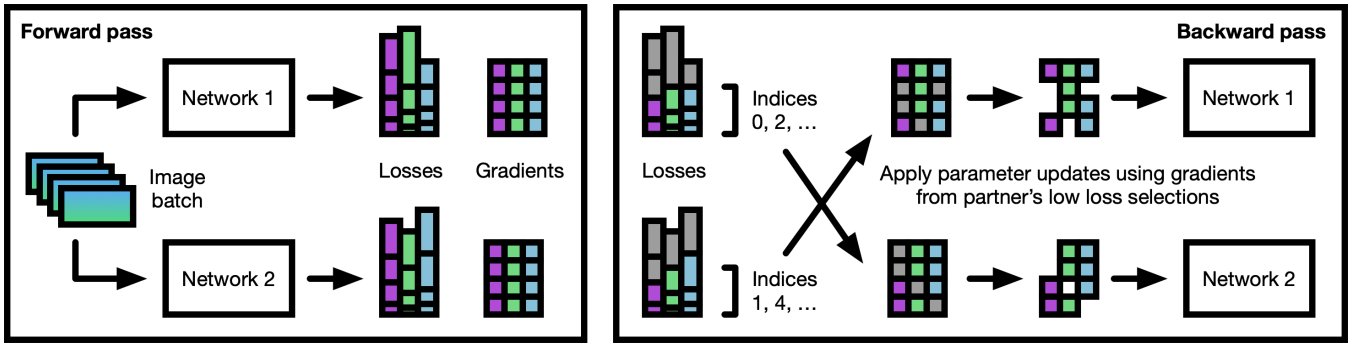

Fig. 3. The modified co-teaching process introduced in [3]. The indices of the lowest loss elements from ordered lists of the three types of object detector loss (losses relating to positives, hard-negatives and bounding box regression) are exchanged between two identical networks and only those lower loss elements are used to provide parameter updates.

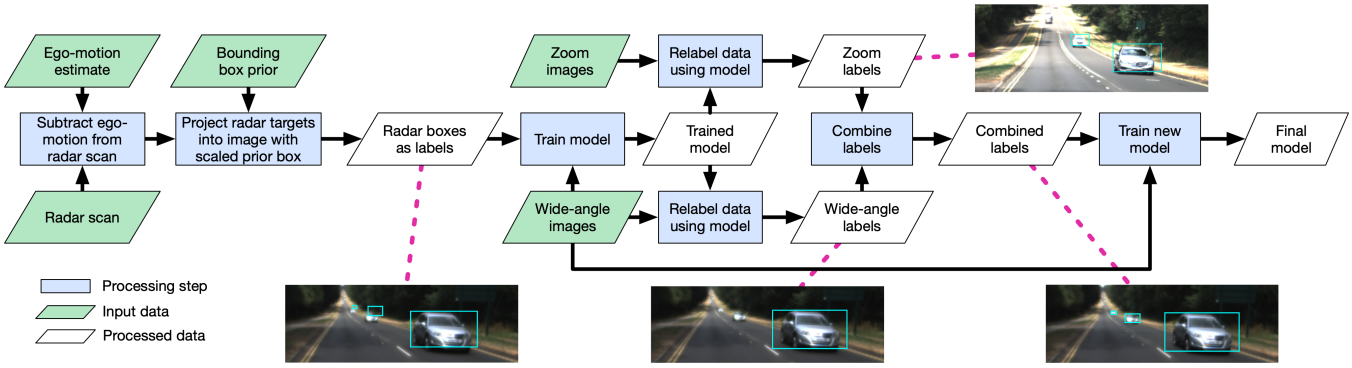

Fig. 4. The relabelling process. The image crops show examples of the labels at the different stages of the process. Note how the original radar labelling process does not account for the slope of the road which results in the bounding boxes being placed above the vehicles. During relabelling this is corrected by detections in the zoom lens image that are combined with the wide image detections to form the final combined label set.

than a number of epochs) for each model (pre- and post-relabelling).

A. Co-teaching Loss Function

When co-teaching is enabled, the overall loss function that is optimised is calculated as

$$\mathbb{L} = \mathbb{L}_P + \mathbb{L}_H + \mathbb{L}_B \tag{1}$$

where $\mathbb{L}_P$, $\mathbb{L}_H$ and $\mathbb{L}_B$ are the losses relating to the positive, hard-negative and bounding box elements respectively. They are defined as

$$\mathbb{L}_i = \sum_{j=0}^{1} \sum_{k=1}^{N_i} L_{j,k}^i \tag{2}$$

where

$$L_{j,k}^i = l_{j,k}^i \mathbb{I}(l_{1-j,k}^i < A_{1-j,t}^i) \tag{3}$$

and $i \in \{P, H, B\}$. The loss for a specific instance k of type i for model j is $l_{j,k}^i$ and $\mathbb{I}$ is the indicator function. The number of instances N_i varies between types (more hard-negatives than positives, for example). The losses $l_{j,k}^i$ are calculated in accordance with [28] with l^P being cross-entropy losses for the positive instances (those anchor boxes matched with ground truth labels) and l^H being cross-entropy losses for hard-negative instances. The bounding box losses l^B are calculated using a smooth L1 loss.

$A_{1-j,t}^i$ is the percentile exponential moving average updated at each step of training and calculated as

$$A_{t+1} = \begin{cases} E(\mathbf{V}, \epsilon), & t = 0 \\ \lambda E(\mathbf{V}, \epsilon) + (1 - \lambda) A_t, & t > 0 \end{cases} \tag{4}$$

where t is the training step, λ is the moving average rate (we use $\lambda = 0.9$) and the i and $1 - j$ have been left off for readability. $E(\mathbf{V}, \epsilon)$ is the loss value corresponding to a given percentile where $\mathbf{V}$ is the vector of all loss values $l_{1-j,k}^i$ for a specific i and ϵ is the required percentile (we use $\epsilon = 1 - \eta$ where η is the expected error fraction).

VII. Dataset

To examine the performance of our method we conduct tests on a dataset comprising over three hours of driving near Oxford, UK, covering urban, sub-urban, highway and rural roads. The dataset includes six individual drives, four used for training and one each for validation and testing. The validation and testing drives follow different routes to any of the drives used in training. In total, after sub-sampling, the dataset includes 8415 examples for training where each

example contains a wide-angle image, a zoom image (which covers the central portion of the wide-angle image), a set of radar targets and an ego-motion estimate. In our case this estimate was obtained using stereo visual odometry [31] (utilising an extra camera not used for network training), although other ego-motion estimation methods could be used. In addition, in order to enable clear evaluation of the learning method, from the separate validation and test drives, 181 validation images and 228 test images have been hand-labelled with 2D bounding boxes around all vehicles. Wide-angle images are 512x1280 pixels in size, zoom images are 960x1280.

VIII. Experimental Results

To examine the performance of our method we perform a range of experiments on our dataset. For all experiments we evaluate using the hand-labelled test set. We use average precision with an IOU threshold of 0.5 and, in line with the KITTI evaluation procedure [32], we exclude labels (and related detections) of vehicles less than 25 pixels high.

In the first set of experiments we show the effect of each part of our relabelling process. We start by simply training a model without any additional processes which demonstrates the issues caused by the noisy radar labels. We then add relabelling but use only the wide-angle images to generate the labels for the second training run. In the next experiment we again use relabelling but generate the labels for the second run by combining labels from both the wide-angle and zoom images. We then run the same experiments again but this time using the co-teaching process in each training loop. Finally, as co-teaching produces two usable models on each training run, in the relabelling process we use both models to contribute zoom labels to the relabelled data. The results of these experiments are shown in Table I and in Figure 5 with some sample detections shown in Figure 6. All results are the average of two runs with those experiments that make use of co-teaching presented as the average over both runs and both models (for a total of four). It can be seen that the combination of co-teaching and relabelling provides a large increase in detector performance. It can also be seen that the use of labels from the zoom image contributes significantly. Our full process yields an improvement of 17 points over the standard training process.

In a further set of experiments, we evaluate the benefit of our method in a scenario where a network has already been trained for the task on a different domain. In this case we pre-train our network on the KITTI object detection dataset. The pre-trained network is then used as the initialisation for both stages of training (i.e. when using only the original radar labels and additionally when using the cleaned labels in the second training run). The results are shown in Table II. It can be seen that while the results in general are higher than those when training from scratch, the relabelling process is still highly beneficial. The effect of the second zoom model is limited to a tightening of the standard deviation rather than an increase in performance.

TABLE I

Vehicle detection performance using different elements of the relabelling pipeline

Relabelling configuration					AP (±1SD)
Co-teaching	**Re-label**	**Wide-angle labels**	**Zoom labels**	**Both zoom models**	
No	No	N/A	N/A	N/A	0.203 (±0.0177)
No	Yes	Yes	No	N/A	0.236 (±0.0085)
No	Yes	Yes	Yes	N/A	0.264 (±0.0042)
Yes	No	N/A	N/A	N/A	0.247 (±0.0431)
Yes	Yes	Yes	No	No	0.280 (±0.0168)
Yes	Yes	Yes	Yes	No	0.363 (±0.0378)
Yes	Yes	Yes	Yes	Yes	**0.377** (±0.0240)

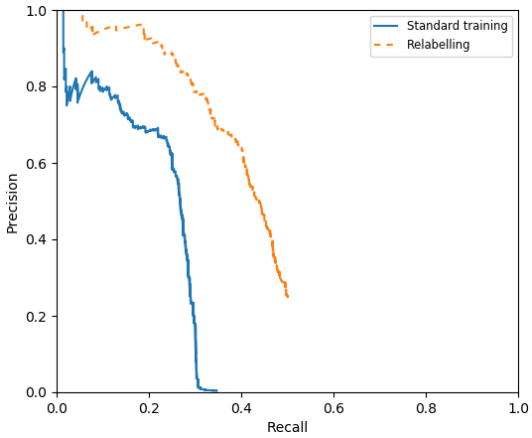

Fig. 5. Average precision curves showing the performance of a model trained using the standard training process against one trained using our full relabelling process.

IX. Conclusions and Further Work

In this paper we have proposed a method for improving the performance of an object detector when using automatically labelled data. The combination of a modified version of co-teaching (introduced in [3]) with the relabelling process using two cameras introduced in this work has been shown to be effective. This allows the use of labelling methods that would previously have been thought to be far too unreliable.

Nevertheless, there are downsides to this method: having to train two models effectively doubles training time. In addition, as is the case with our radar labelling method, biases in the original labels cannot be fully overcome — as the radar cannot determine the tangential velocity of crossing vehicles they are rarely included in the label set which correspondingly reduces the overall detector performance. Similarly, the fixed-size vehicle prior is not suitable for larger vehicles such as buses — we find that while the detector may detect the vehicles, the proposed bounding boxes are too small (and are consequently evaluated as incorrect detections). It is possible that an improvement to our method would be to base the initial boxes on proposals such as those provided by [15].

One facet that we have yet to explore is the effect of

TABLE II

VEHICLE DETECTION PERFORMANCE USING CO-TEACHING AND A PRE-TRAINED MODEL FOR INITIALISATION

Relabelling configuration				AP (±1SD
Relabel	**Wide-angle labels**	**Zoom labels**	**Both zoom models**	
No	N/A	N/A	N/A	0.321 (±0.0168)
Yes	Yes	No	N/A	0.344 (±0.0107)
Yes	Yes	Yes	No	**0.427** (±0.0388)
Yes	Yes	Yes	Yes	0.422 (±0.0192)

training data quantity on the efficacy of our method. It could be that large amounts of additional data provide further performance improvements, or alternatively, in the experiments above our method may be purely compensating for a paucity of data. This could be potentially be explored using the nuScenes dataset [33] which also includes radar data although it does not include zoom lens images which our results indicate contribute considerably to the final performance.

Fig. 6. Example detections from the test set using the model trained with the full relabelling process.

ACKNOWLEDGEMENTS

We gratefully acknowledge the JADE-HPC facility for providing the GPUs used in this work. Paul Newman is funded by the EPSRC Programme Grant EP/M019918/1.

REFERENCES

[1] J. Tremblay, A. Prakash, D. Acuna, M. Brophy, V. Jampani, C. Anil, T. To, E. Cameracci, S. Boochoon, and S. Birchfield, "Training deep networks with synthetic data: Bridging the reality gap by domain randomization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 969–977.

[2] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, and M. Sugiyama, "Co-teaching: Robust training of deep neural networks with extremely noisy labels," in *Advances in neural information processing systems*, 2018, pp. 8527–8537.

[3] S. Chadwick and P. Newman, "Training object detectors with noisy data," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, June 2019, pp. 1319–1325.

[4] Y. Zou, Z. Yu, B. Vijaya Kumar, and J. Wang, "Unsupervised domain adaptation for semantic segmentation via class-balanced self-training," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 289–305.

[5] M. Wulfmeier, A. Bewley, and I. Posner, "Addressing appearance change in outdoor robotics with adversarial domain adaptation," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 1551–1558.

[6] 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," *arXiv preprint arXiv:1711.03213*, 2017.

[7] Y. Chen, W. Li, C. Sakaridis, D. Dai, and L. Van Gool, "Domain adaptive faster r-cnn for object detection in the wild," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3339–3348.

[8] X. Zhu, J. Pang, C. Yang, J. Shi, and D. Lin, "Adapting object detectors via selective cross-domain alignment," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 687–696.

[9] S. Sankaranarayanan, Y. Balaji, A. Jain, S. Nam Lim, and R. Chellappa, "Learning from synthetic data: Addressing domain shift for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3752–3761.

[10] H. Bilen and A. Vedaldi, "Weakly supervised deep detection networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2846–2854.

[11] X. Zhang, Y. Wei, G. Kang, Y. Yang, and T. Huang, "Self-produced guidance for weakly-supervised object localization," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 597–613.

[12] J. Choe and H. Shim, "Attention-based dropout layer for weakly supervised object localization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2219–2228.

[13] C. Ge, J. Wang, Q. Qi, H. Sun, and J. Liao, "Fewer is more: Image segmentation based weakly supervised object detection with partial aggregation." in *BMVC*, 2018, p. 136.

[14] N. Inoue, R. Furuta, T. Yamasaki, and K. Aizawa, "Cross-domain weakly-supervised object detection through progressive domain adaptation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5001–5009.

[15] J. R. Uijlings, K. E. Van De Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *International journal of computer vision*, vol. 104, no. 2, pp. 154–171, 2013.

[16] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *European conference on computer vision*. Springer, 2014, pp. 391–405.

[17] K. K. Singh and Y. J. Lee, "You reap what you sow: Using videos to generate high precision object proposals for weakly-supervised object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9414–9422.

[18] S. Reed, H. Lee, D. Anguelov, C. Szegedy, D. Erhan, and A. Rabinovich, "Training deep neural networks on noisy labels with bootstrapping," *arXiv preprint arXiv:1412.6596*, 2014.

[19] S. Sukhbaatar, J. Bruna, M. Paluri, L. Bourdev, and R. Fergus, "Training convolutional networks with noisy labels," *arXiv preprint arXiv:1406.2080*, 2014.

[20] J. Li, Y. Wong, Q. Zhao, and M. S. Kankanhalli, "Learning to learn from noisy labeled data," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5051–5059.

[21] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[22] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," *arXiv preprint arXiv:1906.05849*, 2019.

[23] D. Pathak, R. Girshick, P. Dollár, T. Darrell, and B. Hariharan, "Learning features by watching objects move," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2701–2710.

[24] D. Arpit, S. Jastrzębski, N. Ballas, D. Krueger, E. Bengio, M. S. Kanwal, T. Maharaj, A. Fischer, A. Courville, Y. Bengio *et al.*, "A closer look at memorization in deep networks," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 233–242.

[25] L. Jiang, Z. Zhou, T. Leung, L.-J. Li, and L. Fei-Fei, "Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels," *arXiv preprint arXiv:1712.05055*, 2017.

[26] M. P. Kumar, B. Packer, and D. Koller, "Self-paced learning for latent variable models," in *Advances in Neural Information Processing Systems*, 2010, pp. 1189–1197.

[27] S. Chadwick, W. Maddern, and P. Newman, "Distant vehicle detection using radar and vision," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 8311–8317.

[28] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[31] W. Churchill, "Experience based navigation: Theory, practice and implementation," Ph.D. dissertation, University of Oxford, Oxford, United Kingdom, 2012.

[32] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 3354–3361.

[33] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," *arXiv preprint arXiv:1903.11027*, 2019.