



Reweighting neural network examples for robust object detection at sea

Becktor, Jonathan Binner; Boukas, Evangelos; Blanke, Mogens; Nalpantidis, Lazaros

Published in:
Electronics Letters

Link to article, DOI:
[10.1049/ell2.12166](https://doi.org/10.1049/ell2.12166)

Publication date:
2021

Document Version
Publisher's PDF, also known as Version of record

[Link back to DTU Orbit](#)

Citation (APA):
Becktor, J. B., Boukas, E., Blanke, M., & Nalpantidis, L. (2021). Reweighting neural network examples for robust object detection at sea. *Electronics Letters*, 57(16), 608-610. <https://doi.org/10.1049/ell2.12166>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Reweighting neural network examples for robust object detection at sea

J. Becktor,[✉] E. Boukas, M. Blanke, and L. Nalpanidis
Department of Electrical Engineering, Technical University of Denmark, Kgs., Lyngby, Denmark

✉ Correspondence

J. Becktor, Department of Electrical Engineering, Technical University of Denmark, Kgs. Lyngby, Denmark.
Email: Jbibe@elektro.dtu.dk

Deep neural networks have had profound significance in addressing visual object detection and classification tasks. However, though with the caveat of needing large amounts of annotated training data. Furthermore, the possibility of neural networks overfitting to the biases and faults included in their respective datasets. In this work, methods for achieving robust neural networks, able to tolerate untrusted and possibly erroneous training data, are explored. The proposed method is shown to improve performance and help neural networks learn from untrusted data, provided a thoroughly annotated subset.

Introduction: Deep neural networks provide strong tools for solving problems such as visual object detection [1, 2]. However, such capabilities often come with the requirement of considerable amount of carefully annotated training examples. Robotic systems working in real-world environments, training examples are difficult to acquire and in particular for rare events. On top of this, precise manual annotation is a tedious and error-prone process. Thus, automating the annotation process or making the training of deep neural networks robust against mistakes in the provided annotations would significantly improve the practical applicability of deep neural networks in robotic applications.

This letter explores a possible method for an automatic annotation framework. The approach presented enables the creation of more robust neural networks by following a carefully designed training pipeline, a process that can tolerate training on data with partially missing or incorrect annotations. Given a dataset \mathcal{S} of which \mathcal{S}_u may comprise questionable annotations, we wish to convert part of \mathcal{S} to a trusted set by thorough manual re-annotation. This subset referred to as the reweight-set \mathcal{S}_R . The method used for reweighting data was introduced in [3]. Given a reweight set, they computed weights to be applied on each element in the training data. Ideally, this method makes it possible to obtain networks with higher performance, while only having to manually annotate a portion of the data. Furthermore, the method ensures that incorrect labels have less of an impact on training and generalization. Simultaneously, this enables better generalization on the correct annotations through the reweighting of each sample.

Furthermore, this letter employs a sampling method of input images which ensures an uniform size of annotation bounding boxes. The sampler crops the input images and resizes each crop.

The contribution of this work is twofold: (i) It applies the reweighting scheme introduced in [3] and verifies that it works in a challenging real-world maritime setting; this is done by introducing a pipeline that allows us to generalize on a wider range of samples without having access to a large trusted dataset; and (ii) it establishes a method that contributes to an automatic annotation framework that allows generalization when existing data \mathcal{S}^{exist} is extended by new automatically annotated data \mathcal{S}_u^{new} , and a subset of that is manually annotated \mathcal{S}_R^{new} .

Related work: This work focuses on the use of deep neural networks to achieve robust object detection at sea. Previous approaches have been along the lines of [4] where detection of small surface vessels was achieved by using scale invariant feature transform (SIFT) along with a bag-of-features approach. The work of [5] explored the use of neural networks, comparing using RGB, long wave infrared (LWIR), and near infrared (NIR) data within the Resnet-50 RetinaNet. The importance of weighting training data in a maritime settings is yet to be explored, in the wider field it is a well-studied topic. More precisely, importance sampling [6] assigns weights to samples so that distributions match. Focal

loss [7] introduces a soft weighting scheme that focuses on harder examples. Hard example mining [8] down-samples the majority class and the hardest examples are exploited. Similarly, AdaBoost [9] finds out harder examples while training. However, it is not always beneficial to prefer hard examples as these might be outliers, noisy, or even wrong annotations. Another common issue is class imbalance, several studies have also been made here such as dataset re-sampling [10]. The work of [3] introduces a scheme to weight each training sample according to a smaller highly trusted reweight-set. The need for robustness of neural networks is addressed in [11], where the authors constrain the Lipschitz constant of a network to be lower than 1 by constraining each layer's weight matrices to be orthonormal. Furthermore, [12] explores how training neural networks under a strict Lipschitz constraint is useful for provable adversarial robustness. In our previous work in [1], we explore the use of Lipschitz constrained networks in a maritime setting to ensure robustness of predictions.

Method: In this work, we are adapting the theory on learning to reweight examples introduced in [3] and enhance it with a custom data augmentation technique.

Reweighting: The introduction of the *reweight-set* enables training of networks with questionable data. We define a loss function $f(\theta; \mathcal{S})$ given the model f with parameters θ and trained on the data \mathcal{S} . In the following, we will apply an online weighting ϵ on $f(\theta; \mathcal{S})$ and obtain an optimal weighting of each sample in the training loss $f(\theta; \mathcal{S}_u)$. The online weighting ϵ is also referred to as perturbation in [3]. We compute the optimal weighting of the training loss by finding the gradients of $f(\theta; \mathcal{S}_u)$ with respect to parameter vector $\theta(\mathcal{S}_R)$ of the model that is found with forward pass of the reweight set. This is obtained as follows, where i is the iteration,

$$f_{i,\epsilon}(\theta) = \epsilon_i f_i(\theta), \quad (1)$$

where each training example of the original model is perturbed with ϵ at time-step t , described as following when considering SGD (stochastic gradient descent),

$$\hat{\theta}_{t+1}(\epsilon) = \theta_t - \left(\alpha \nabla_{\theta} \sum_{i=1}^n f_{i,\epsilon}(\theta) \right) \Bigg|_{\theta=\theta_t}. \quad (2)$$

To find the optimal ϵ that minimizes the meta-loss we apply,

$$\epsilon_t^* = \arg \min_{\epsilon} \frac{1}{M} \sum_{i=1}^M f_i^m(\theta_{t+1}(\epsilon)). \quad (3)$$

This is quite time consuming, therefore, a gradient step with regard to the meta-loss ϵ^m is taken instead.

$$u_{i,t} = - \left(\eta \frac{\partial}{\partial \epsilon_{i,t}} \frac{1}{m} \sum_{j=1}^m f_j^m(\theta_{t+1}(\epsilon)) \right) \Bigg|_{\epsilon_{i,t}=0}, \quad (4)$$

where η is the descent step size for ϵ .

This is an approximation of the optimal ϵ at time-step t , and thereafter the weights of the meta-model are updated. This is then followed by a forward pass and backwards pass on the Reweight-set.

By finding the gradients of the updated meta-model with respect to ϵ , and by rectifying the output, we get the weights we can apply to the loss of each sample in the original batch. The weights are then normalized to sum to one for a more consistent parameter update.

Data Augmentation: For more efficient training and better use of the reweighting scheme, a data cropping augmentation is introduced. In the considered marine environment, the horizon line can be easily detected (cf. Figure 1). Thereafter, images of 360×360 pixels are cropped out along the horizon line. In addition, two full-height square patches are extracted from the original image and all the newly cropped images are scaled to 384×384 pixels. The cropped images are randomly sampled making sure that any existing labels are selected at least once; thus an issue previously faced such as an overwhelming amount of images without boats and buoys is alleviated. Given hundreds of training iterations, it can be assumed that the network has seen all annotations

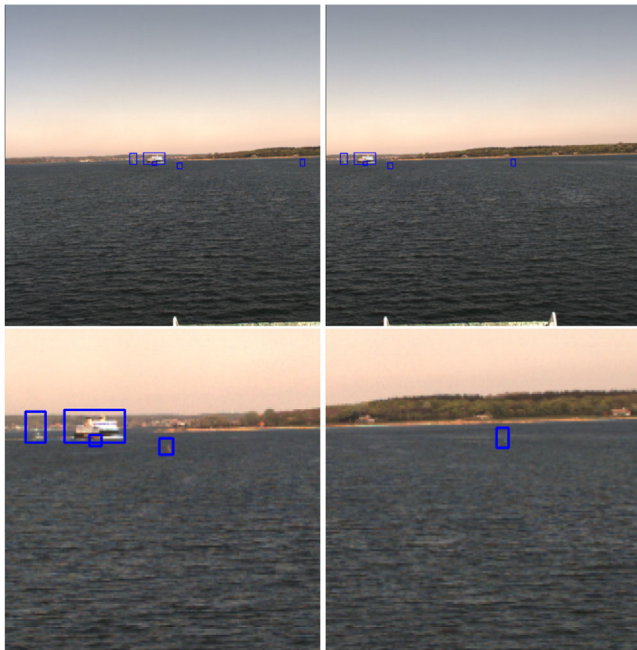


FIGURE 1 Example crops of an image resized to 384×384 pixels

multiple times. Introducing data-cropping is a necessary part of our pipeline to reduce the dimensions of input images and to normalize the size of all annotations. As we reduce the pixel size of the image from 1920×1080 to 384×384 , we reduce the size of the input by a factor of $(1920 \cdot 1080)/384^2 = 14.0625$ which has a tremendous impact on the speed of a single forward pass of the network. This allows us to batch multiple different crops from different images which allows the networks to converge faster.

Experimental evaluation: In this section, we present the experimental set-up and the results of our approach.

Datasets: The work of [5] introduced the dataset which is used as the baseline for the experimental evaluation of our work. Additionally, a newly collected set of samples have been classified using the model introduced in [5], with a very low confidence requirement, resulting in several miss-classified labels. The available sets of data have been used to create four different blends comprising: (i) 100% and 0%, (ii) 95% and 5%, (iii) 85% and 15%, of *trusted* and *untrusted* data, respectively, where trusted data stem from the original dataset and untrusted data come from the low-confidence classified dataset.

Experiments: For each dataset introduced, two networks were trained. The network architecture in use is a downscaled version of the RetinaNet with a ResNet 18 backbone [13]. The size of the network is reduced by a factor of 100 to increase the speed and the memory requirements of training. While shallower networks – like the one we present here – result in inferior performance compared to fully parameterized ones, the performance trends are consistent between the two.

Training: All models were trained with the AdamW-optimizer [14] for 24 h on a Tesla V100 GPU with 32 GB of memory. We used a batch size of 32 with a step scheduler that after each 50 epochs reduces the learning rate by a factor of 10.

Results and discussion: The results in Table 1 provide a positive argument for the case of reweighting. When untrusted data is introduced to the datasets, the reweighted networks on average reach a higher mean average precision (mAP). In our considered data blend (i), no new data is introduced. Looking at the first plot in Figure 2 the standard network performs slightly better when predicting boats than the reweighted network. This can be a hint of the Reweight-set not being expressive enough. Though this can also be a testament to the base dataset being properly annotated. The second plot in Figure 2 uses our data blend (ii), that is, when 5% of all labels are untrusted. Here, it is clearly visible on both boats and buoys, that the reweighting mechanism allows for wider gen-

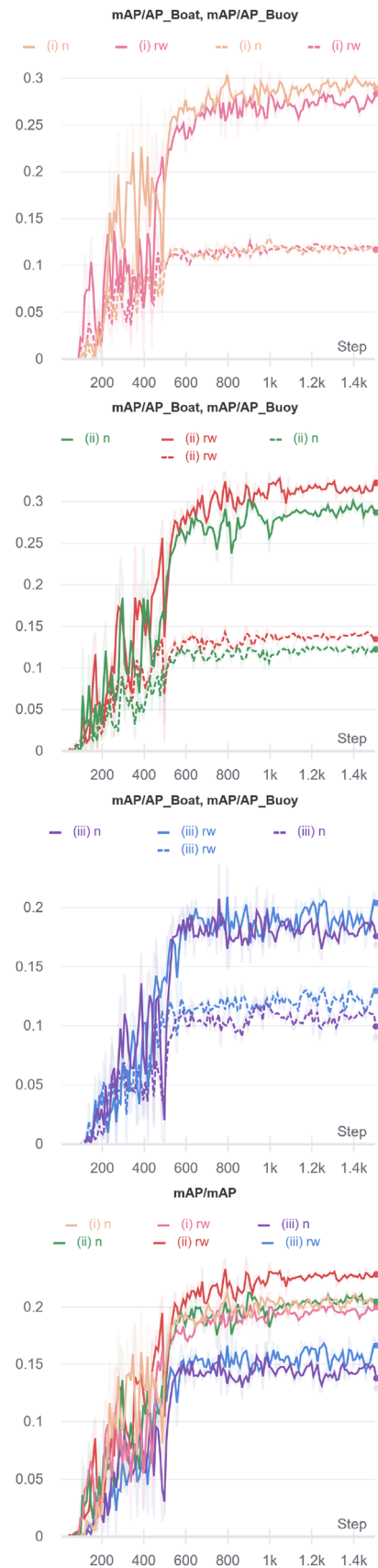


FIGURE 2 Comparison of the average precision (AP) of boats and buoys for the different datasets. First plot is the original data blend (i) where beige is the normal network and pink the reweight. In the second plot, data blend (ii): green is normal, red is reweight with 21k annotations 5% untrusted labels. In the third plot, data blend (iii): blue is reweight, purple is normal with 9k labels and where 15% are untrusted. Finally a mean average precision (mAP) comparison of all networks where the colours are persistent from the previous plots

Table 1. AP for boats (i) and buoys (ii), and mAP for network (iii)

AP Buoy	(i)	(ii)	(iii)
Normal	0.120 ± 0.006	0.124 ± 0.004	0.108 ± 0.009
Reweight	0.119 ± 0.005	0.135 ± 0.005	0.126 ± 0.008
AP Ship	(i)	(ii)	(iii)
Normal	0.281 ± 0.014	0.302 ± 0.011	0.180 ± 0.012
Reweight	0.285 ± 0.012	0.333 ± 0.008	0.198 ± 0.015
mAP Ship & Buoy	(i)	(ii)	(iii)
Normal	0.200 ± 0.010	0.205 ± 0.008	0.146 ± 0.010
Reweight	0.206 ± 0.09	0.228 ± 0.006	0.163 ± 0.011

eralization using the untrusted data. Comparing the normal network for data blends (i) and (ii), it can be noted that they have practically the same mAP showing that they do not benefit from the addition of untrusted data. The third plot in Figure 2 uses our data blend (iii); the amount of labels used is reduced from 20k to approximately 9k. Here, 15% of all labels are untrusted and the reweighting again does a better job, though not to the extent expected. The reweight network has a higher mAP on both fronts, yet we would expect the difference to be larger when the proportion of untrusted data is higher. This could possibly be caused by the network not having enough information to generalize on and therefore the reweighting mechanism cannot weight the information enough to make a bigger difference. This can further be seen when plotting the mAP of all the networks, see the fourth plot of Figure 2, where (iii) performs 40% worse.

Conclusion and future work: The results present a clear benefit when a reweight-set with reweighting mechanism is implemented. The suggested approach achieves a higher mAP on all datasets with questionable annotations. The importance of which samples to include in the reweight-set is a subject of further research. Our initial tests have all been on scaled down versions of the RetinaNet [7] with promising results. We leave the experimental evaluation of the fully parameterized neural networks to future work. The introduced data augmentation technique was shown to retain the performance of a network using larger images whilst training much faster. Furthermore, as increase of the batch size of our network was possible, there was faster and smoother convergence. The approach demonstrated its ability to tolerate questionable annotations in a dataset when a small trusted Reweight-set was introduced. Given these results, the reweighting mechanism proves itself as a potential method to use for an automatic annotation framework.

Acknowledgement: This research is sponsored by the Innovation Fund Denmark, The Danish Maritime Fund, Orients Fund, and the Lauritzen Foundation through the ShippingLab project, grant 8090-00063B.

© 2021 The Authors. *Electronics Letters* published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

Received: 2 March 2021 Accepted: 3 March 2021

doi: 10.1049/ell2.12166

References

- Becktor, J. et al.: Lipschitz constrained neural networks for robust object detection at sea. In IOP Conference Series: Materials Science and Engineering, Vol. **929**, (2020)
- Schöller, F.E.T. et al.: Vision-based object tracking in marine environments using features from neural network detections. In: IFAC World Congress, Berlin, Germany (2020)
- Ren, M. et al.: Learning to reweight examples for robust deep learning. In 35th International Conference on Machine Learning, ICML 2018, volume 10, Stockholm, Sweden, July 2018
- Dulski, R. et al.: Detection of small surface vessels in near, medium, and far infrared spectral bands. In Electro-Optical and Infrared Systems: Technology and Applications VIII, Prague, Czech Republic (2011) <https://www.spiedigitallibrary.org/conference-proceedings-of-spie/8185/81850U/Detection-of-small-surface-vessels-in-near-medium-and-far/10.1117/12.898272.short?SSO=1>
- Stets, J.D., et al.: Comparing spectral bands for object detection at sea using convolutional neural networks. *J. Phys. Conf. Ser.* **1357**, 12036 (2019)
- Kahn, H., Marshall, A.W.: Methods of reducing sample size in monte carlo computations. *J. Oper. Res. Soc.* **1**, 5, (1953)
- Lin, T.Y. et al.: Focal loss for dense object detection. In: Proceedings of the IEEE International Conference on Computer Vision 2017, pp. 2999–3007 (2017)
- Malisiewicz, T., Gupta, A., Efros, A.A.: Ensemble of exemplar-svms for object detection and beyond. In: 2011 International Conference on Computer Vision, pp. 89–96, (2011)
- Freund, Y., Schapire, R.E.: A Decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.* **55**(1), 119–139 (1997)
- Chawla N. V., Bowyer K. W., Hall L. O., Kegelmeyer W. P. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research* **16**, 321–357 (2002) <http://doi.org/10.1613/jair.953>
- Szegedy, C. et al.: Intriguing properties of neural networks. In: 2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings, (2014)
- Anil, C., Lucas, J., Grosse, R.: Sorting out Lipschitz function approximation. In: 36th International Conference on Machine Learning, ICML 2019, Long Beach, California, USA, Vol. **2019** June, pp. 432–452 (2019)
- He, K. et al.: Deep residual learning for image recognition. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, (2016)
- Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In International Conference on Learning Representations, New Orleans, USA (2019)