Domain Adaptive Wheat Head Detection

Shreya Chawla
u7195872

Prithvi Chandregowda Venkat
u7095159

Abstract

The project aims to implement Wheat head counting by object detection in an efficient way. It addresses the efficiency problem observed in previous methods which affects their scalability and cost-effectiveness for practical usage. Our problem of domain adaptation is to generalize well enough to overcome the domain gap in target and source distributions. To achieve this, a smaller version of the existing state-of-the-art model YOLOv5 is used called YOLOv5s. It has 16.5 B FLOPs compared to 205.7 B FLOPs in YOLOv5s. But the mAP and IoU scores are not comparable. We propose a new pipeline to overcome this. It makes use of pseudo labelling and weighted bounding boxes to improve upon the baseline. The motivation behind this pipeline is to exploit the different distributions learnt by different models on the small GWHD dataset. Additionally, since our network is a smaller variant of the original model, we can train several models simultaneously using cross validation. The efficiency is of our model is still higher than the current state of art in terms of FLOPs count with some loss in accuracy. After implementing and conducting experiments on the proposed method, we observe that the proposed pipeline utilized only 75% of the computational costs utilized by the dataset benchmark Faster-RCNN model. Also, our model achieves the test result of 0.635 mAP@0.5, which is very close to the Faster-RCNN model's test result of 0.638 mAP@0.5 precision and better than the original model at 0.616 mAP@0.5. We were able to achieve a remarkable WDA score of 0.601, which is slightly better than original model.

1. Introduction

Wheat is one of the most important staple crops of the world. This project aims to implement object detection on wheat crop in the field for counting wheat heads. Our paper attempts to solve this problem in a computationally efficient manner while maintaining performance for better scalability in computationally constrained devices like mobile phones and drones.

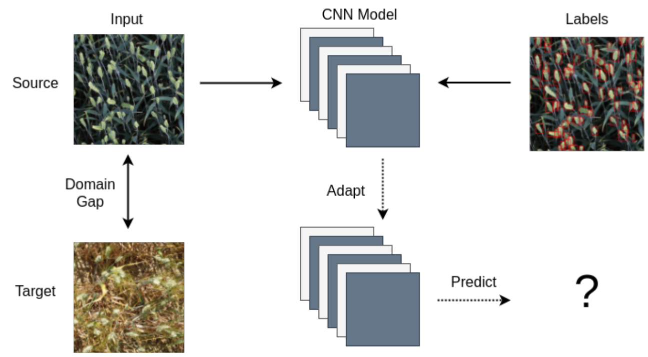


Figure 1. Domain adaptation problem statement

As the world population is increasing drastically, according to the UN population estimate there will be 10 billion people living on the planet earth by 2050. There will be very high increase in demand of food and resources. In order to increase the food production while trying to minimize the input, there is need to adapt advance computing technologies for better analysis and monitoring. This field has recently gained a lot of attention among researchers. Precision agriculture is an evolving field which takes advantage of the latest advanced computing technologies to minimize the inputs required, to improve the crop quality and increase yield per meter square to cater the increase in demand. Deep learning based computer vision techniques can help with detection from target crop images extracted from multiple sources. This has shown tremendous advantages over the traditional methods applied in agriculture [9].

Wheat head detection can be used to evaluate yield of the season, and perform phenotype research to access the variations and hybridisation of crop species. Estimating pertinent wheat traits including head population density and head characteristics such as health, size, maturity stage, and the presence of awns can help in this. This research can have severe implications in advancement of agriculture. It can also be used to assess ear density, early detection of diseases and pest, monitoring crop health, water stress, soil fertility and maturity of crop more effectively for farmers to keep track of their yield. This ensures prevention and fast management in adverse conditions [15] [7] [21].

Counting wheat heads is an important but long and tedious task traditionally performed manually from images of outdoor fields. Annotating a wheat dataset is a very time consuming and expensive affair. Hence, we introduce out method which is not only effective in computational power and memory, but also trains on unseen data for semi-supervised learning. This can be further extended for even bigger unannotated datasets.

Wheat has many species across the world [1]. Our task is to detect wheat ears of all possible species. Hence, the source and target distribution of the wheat crop images would not come from the same marginal distribution but the task performed on both is same - wheat-head detection. Hence, this is a domain adaptation problem as shown in Fig. 1. In this work we attempt to reduce the domain gap between these two distributions for higher robustness and accuracy. High variability in observational conditions, genotype differences, development stages, and head orientation makes digital detection of wheat head a challenge for computer vision. Further, factors like overlapping heads, presence or absence of barbs, and blurring due to motion or wind, makes this task more complex.

One of the best object detector algorithm is Faster RCNN [20]. It owes its high performance to region proposals. But that comes at cost of its inference time being very large due to a large number of floating point operations in the network. Rather than using this two-stage detector, a one-stage object detector like YOLO [3] can provide results in real-time with a similar precision. Our technique is based on Yolov5 [3] which can provide the optimal speed and accuracy of object detection. We show that improvements in the pipeline can boost the results.

Our model depicts a significant level of efficiency achieved in terms of FLOPs count as compared to the dataset baseline [20]. The Global Wheat HEAd deTection (GWHD) 2021 dataset [1] version is used to implement this strategy. After implementing and conducting experiments on the proposed method, we observe that the network utilized only about 75% of the computational costs utilized by the RCNN model in terms of number of FLOPs. Also, it achieves the test result of 0.635 mAP@0.5, which is better than the baseline model's test result of 0.616 mAP@0.5. On the WDA metric, we achieved 0.601 performance. Upon comparing with the state-of-the-art methods, we see that our method is comparable because with a much smaller network, we are able to achieve very close results.

2. Related Work

Object detection models can be categorized into one-stage and two-stage detectors. One-stage object detectors like YOLO [19] and SSD [14] use a single network to be trained and optimized end-to-end. In first stage in two-stage detectors is a sparse set of proposals are generated to remove maximum negative object location proposals while preserving the location proposals for actual object. The second stage predicts if the remaining proposals are objects (foreground) or background classes. R-CNN [5], Fast R-CNN [4], Faster R-CNN [20] all use CNN to employ this technique for improved performance and sequentially improved inference speed. Recently, a small network called EfficientDet [22] by Mingxing Tan et al proposes a bi-directional feature pyramid network which achieved the state-of-the-art with lesser inference time but a large train time.

Previous works in crop analysis with object detection, used smaller datasets like ACID [18] and SPIKE [16]. ACID dataset used indoor images captured in constrained environment to train thus proving to be insufficient for handling variations. Due to small dataset with less or no diversity in phenotype, these methods could not be generalized to all wheat crops from the world. Hence, these methods were not robust and had low accuracy for a new species of wheat. The previous methods did not address the image blur due to wind or camera shake. They also use huge networks to obtain results. However, in our method we address this issue by using training and testing data augmentations.

Keyhan Najafian et al [17] introduced a semi-self-supervised learning approach for wheat head detection. They used extremely small number of labeled samples from one video and then manually contoured the foreground wheat heads. They extracted different backgrounds from videos, then combined them to develop a strongly annotated dataset. Unlike their method, we did not manually annotate our images. We used mosaic [8], cutmix [23], mixup [24], scale, rotate, flip, shear, and cutout [2] augmentation among many others to stretch our limited data.

Chengxin Liu et al [13] proposed a Dynamic Color Transform (DCT) block which can be added to any network with negligible additional cost. They trained DCT with the network loss in which they use color cues that benefitted wheat head detection. However their method worked very poorly for blur images and their method is very similar to color channel augmentations that are already implemented. Hence, we chose to train four networks on different images with color augmentation.

We perform pseudo labeling [11], an unsupervised technique, where the model is retrained with a fusion of the training and testing data, where the predictions of our model are treated as pseudo labels on the test set.

3. Dataset

Global Wheat HEAd deTection (GWHD) dataset 2021 [1] is the largest dataset for wheat head detection. GWHD is part of WILDS collection [10] by Stanford and was a part of ECCV - CVPPA2021 challenge. It consists of 6500 images with 275,000 wheat heads collected from 12 countries from over 3 continents. The dataset has 47 domains or sub-datsets, where images from one location are acquired with a single sensor while mixing several development stages. Due to several development stages, high variation in color, shape, and orientation are observed across all the domains. The domains are highly unbalanced with high diversity useful to test the model's robustness. Only 18 domains in the Dataset are labelled contributing to 3655 images. Hence, the annotated subset of dataset is split into train, test and validation sets. The distribution of sub-datasets or domains in the labelled dataset (Fig. 2) shows that the data is highly unbalanced. From here on, we refer to the annotated dataset as dataset.

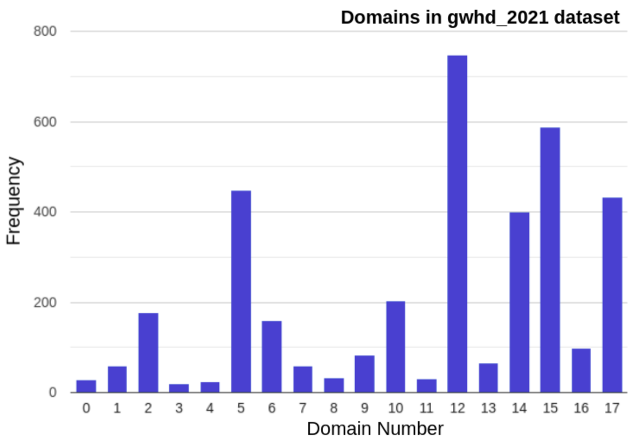


Figure 2. Distribution of domains in labelled dataset.

3.1. Pre-Processing

Upon analysis of the dataset, we found a mismatch in the labelled meta data and number of images. We find that there are 2 duplicated for two images. One of these redundant entries in the dataset when visualized, we found that the bounding boxes for these images does not match the image at all as shown in Fig. 4. These mismatched rows are removed to get a clean dataset. As seen in Fig. 2, the distribution is unevenly divided. The dataset is split into test and train with 655 images from 8 domains and 3000 images from the remaining each. The train set is split into stratified K-folds due to imbalance. The labels and images are saved in accordance to the requirements of YOLOv5 [3]. The bounding box strings following the notation $(x_{start}, y_{start}, x_{end}, y_{end})$ are encoded to the format $(x_{center}, y_{center}, width, height)$ which is a pre-requisite to train YOLO model. This has been shown in Fig. 3. Since we only want to predict wheat heads, the number of classes is set to 1 and denoted by '0'. There strings are appended to the label text files in $(0x_center, y_center, width, height)$ fashion.

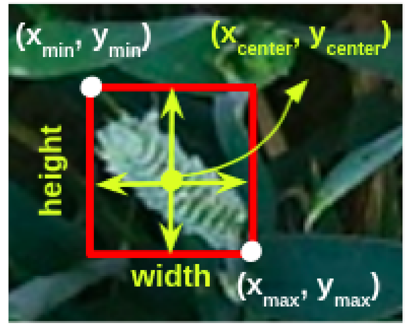
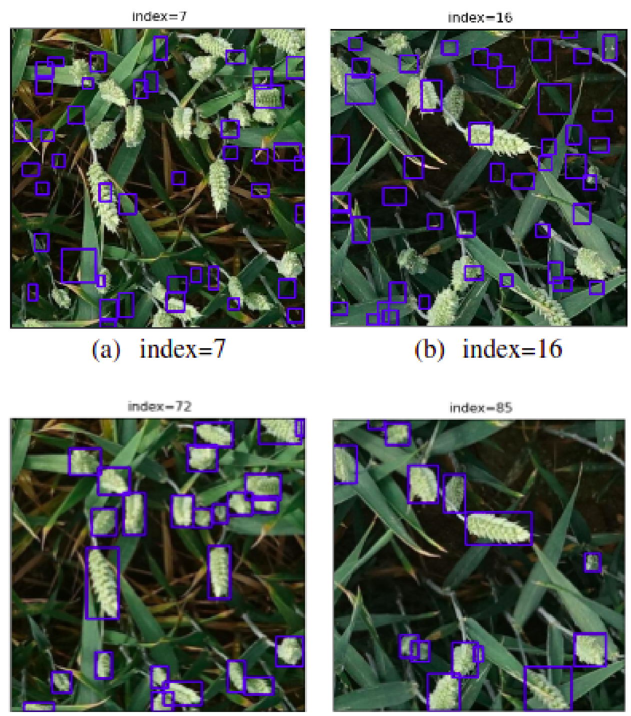


Figure 3. The traditional notation in white follows $(x_{start}, y_{start}, x_{end}, y_{end})$. In yellow is the $(0x_center, y_center, width, height)$ notation accepted by YOLO. We convert bounding box embeddings from white to yellow.



(a) index=7 (b) index=16

(c) index=72 (d) index=85

Figure 4. Visualization of the duplicate values observed during pre-processing.

3.2. Data Augmentation

Data Augmentation expands our small dataset for higher robustness to images obtained from various environments. It provides more images to train on thus improving the performance by reducing the memorization and increasing the learning of patterns. In addition to the default augmentations of YOLOv5, we perform up-down flip, left-right flip, and rotation. We also perform Gaussian blur to boost the prediction when image is blurry due to wind or hand shake. The rotations and flips help in making the model rotation-invariant. As the dataset consists of wheat at different growth stages, the default scaling augmentation helps generalization of model. These different maturity levels lead to high variation in color and to ensure that our methodology can work independent of color, the default HSV color scaling raises the correct estimation probability.

4. Method

4.1. Baseline Architecture

We adopt the state-of-art object detector YOLOv5s [3] as our baseline. YOLOv5 is the next iteration of the popular YOLO research. The architecture of YOLOv5 is based on YOLOv3 [19]. It is faster and more accurate than YOLOv3. It introduces concept of Auto-anchor which allows the model to automatically evolve new anchors to fit the custom data with the same loss as used to train data if the existing anchors fall below the matching threshold. This increases the performance results of the network. The YOLOv5 [3] heavily uses scaling, color space augmentations at its core to improve it's results. The Pytorch implementation [8] of [19] by the same authors, introduced Mosaic augmentation technique that randomly selects and places four images and then applies the above augmentations together on all four images. This has also been used in [3] as it improves robustness of the network. The reason to not choose YOLOv5s6 is that it performs well for large object detection with added memory costs. Since our object to be estimated is not large, we chose to use YOLOv5s. Although YOLOv4 [6] performs slightly better than YOLOv5, we chose YOLOv5 as it has better scalability and inference time at little a loss in mAP.

A limitation of YOLOv3 implemetation [8] is that the input image was scaled up and passed it as a batch during training, thus using a lot of memory. In YOLOv5, rather than changing size by scaling up or down the actual image, the image size is kept the same as input by cropping. This gives an advantage to scale every image in the batch differently thus increasing variation while not requiring extra memory to train very large images. Our model is based on smallest in the family - YOLOv5s network[3] and we compare our results with it as baseline object detector. We use the one-stage object detector YOLOv5 because of better inference time.

4.2. Novel Pipeline

The data is split into 3 sets - train, validation and test as explained earlier. Test consists of 655 images of domains - 0, 1, 2, 3, 6, 8, 9, and 16. This data is kept aside to test our model's performance on. The remaining 3000 images are split into 4 folds to get train and validation sets of 2250 and 750 images each. Thus, the split ratio of train:validation:test set is 61.5:20.5:17.9 approximately. The model is trained for all the four splits and then the ensemble of their predictions is calculated with weighted bounding box fusion technique. Next, pseudo labels are calculated for the unseen test set and the model is trained for another 15 epochs with this semi-supervised technique. It is an effective technique for small dataset like ours to predict more correct labels. The confident predicted test labels are added to the training process. Finally all the predictions are voted using ensemble technique to exploit the differently learnt models. The training of the model is guided by focal loss. This is shown in Fig. 5.

4.3. Evaluation Metrics

There are three metrics that are employed in the proposed method. Two of are based on the Intersection over Union (IoU). In the proposed method, the IoU threshold value t is taken to be 0.5 to compare a ground truth box (GTB) with predicted box (PB). The metrics use the following definitions:

i) tp (true positive) - GTB matched with one PB
ii) fp (false positive) - PB that matches no GTB
iii) fn (false negative) - GTB that matches no box

The evaluation matrics are:

i) **WDA** - Weighted domain accuracy is a proposed as a newly metric [1]. This metric is used to check the accuracy of network across all the domains computed to understand the performance of the model. The detection accuracy $AI_d(i)$ for an image *i* belonging to domain *d* is defined as:

$$AI_d(i) = \frac{TP}{TP + FN + FP} \tag{1}$$

TP, FN and FP are the number of true positive, false negative and false positive respectively found in image *i*. WDA is computed as the weighted average of all domain accuracies:

$$WDA = \frac{1}{D} \sum_{d=1}^{D} \frac{1}{n_d} * \sum_{i=1}^{n_d} AI_d(i) \tag{2}$$

ii) **mAP** - Mean Average Precision is a standard evaluation criteria for object detection tasks. It is also calculated with confidence level for each bounding box to check performance of the model on test data calculated at threshold *t*=0.5. Since our data has only one class - wheat, hence the mean average precision (mAP) is same as average precision (AP), which is computed as:

$$mAP = AP = \frac{TP}{TP + FP} \tag{3}$$

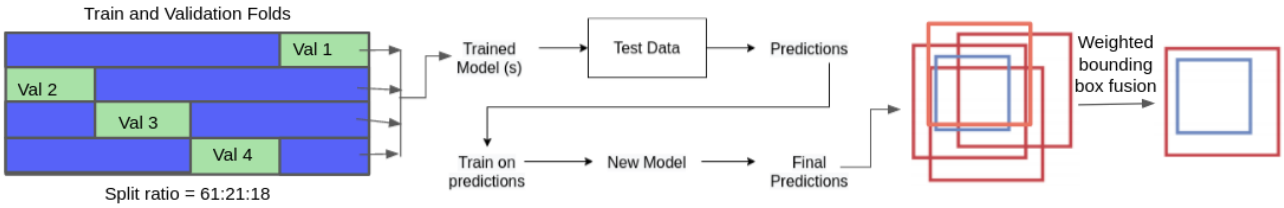


Figure 5. Pipleline for training and testing consists of first training on data for 30 epochs then predicting pseudo labels for test data. A new model is trained on the confident predictions for 15 more epochs, The results of these 4 new model(s) are combined using weighted bounding box ensemble technique where each bounding box is associated with a weight and the final prediction is found accordingly.

TP, FN and FP are the number of true positive, false negative and false positive respectively across all images.

iii) **FLOPs** - It is defined as number of floating point operations performed. It's employed to measure the efficiency of model.

5. Results

5.1. Experiment Setup

The baseline YOLOv5s is trained on the cleaned data. The pre-trained model is loaded and trained for 30 epochs where images of batch size 8 are downsampled to 800. Next, to evaluate its performance, detection algorithm is deployed for downsampled test set images of size 800 on the previously fine-tuned model. Test time augmentations like scaling and color scaling were applied and the results were calculated at a threshold of 0.5. Due to computational resource constraint, we could only train for 30 epochs.

For training the 4 fold cross validation models, the same hyperparameter values gave the best result. For each fold, the pseudo labels of validation set are obtained for the respective best learnt model using detection algorithm at threshold of 0.5, with test time augmentation on downsampled images. Afterwards, a new model is trained for 15 epochs with additional augmentations to finetune as in the default YOLOv5. The bounding boxes are converted back to the $(x_{start}, y_{start}, x_{end}, y_{end})$ notation, that is convert from yellow to white notation from Fig. 3. These new models are then combined and evaluated on the test set where using the last fold model, we re-run pseudo labelling on test set for 15 epochs and finally detect the labels on this newly trained model. All the detection is computed for threshold of 0.5 finetuned with default values. All the images in all the processes are downsized to 800 from 1024. All the training is done for 30 epochs, and re-training after pseudo labelling for 15 epochs.

5.2. Quantitative Results

We compare our model with the original YOLOv5s [3] model upon which we build our pipeline. We also compare our results with Faster R-CNN [20] network which is the baseline for the dataset paper [1]. The quantitative analysis of our model with 2 baselines are presented in Table 1. From the table, it can be inferred that the the dataset baseline network deploys 88 x 10^9 FLOPs, whereas, our network deploys only 66 x 10^9. Although, the deployment cost of our model is four times that of the original model [3], however it is still about 25% less than the dataset baseline [20]. Therefore, there is a significant reduction in computational costs as our pipeline requires approximately only about 75% of the computational resources required by the teacher network. Also, in terms of performance, our model is able to preform better than the initial model. The pre-trained YOLOv5s model does not perform very well on this data as it was trained earlier on COCO dataset [12] which has 80 classes compared to only one in GWHD dataset. Due to this discripancy, the model might be predicting many false positives.

Method	mAP@0.5	WDA	FLOPs (B)
Faster-RCNN [20]	0.638	0.608	88
YOLOv5s [3]	0.616	0.599	16.5
YOLOv5x [3]	0.689	0.700	205.7
Ours	0.635	0.601	66

Table 1. Evaluation results and deployment cost of our model compared to SOTA models in terms of WDA, mAP@0.5 and FLOPs (B: 10^9)

5.3. Qualitative Results

The qualitative results are presented in Fig. 6. As can be seen from the image, the original network detected several overlapping bounding boxes however in our model, this problem is comparatively reduced. This is the result of weighted bounding box algorithm which fuses the predictions of all models to get a more robust estimate.

In some cases our model fails to distinguish between plant and wheat-head. We observe incorrect bounding boxes being predicted for images with similar colors of wheat head and the plant body. In Fig. 7 several incorrect predictions are made where there is a cluster of yellow-green leaves. This confuses the model to believe it to be wheat head leading to false positives and intersected bounding box predictions. We observe that our model has a few duplicate predictions compared to the baseline model. This shows that our model is able to estimate better labels due to ensembling of results of the four trained models that learn different features as the source and target predictions (train and validation set) are different for each model. This helps us optimally leverage the diversity in data.

(b) Baseline YOLOv5s result (a) Our result

Figure 6. Visual tests of performance of our model compared to model baseline YOLOv5s [3].

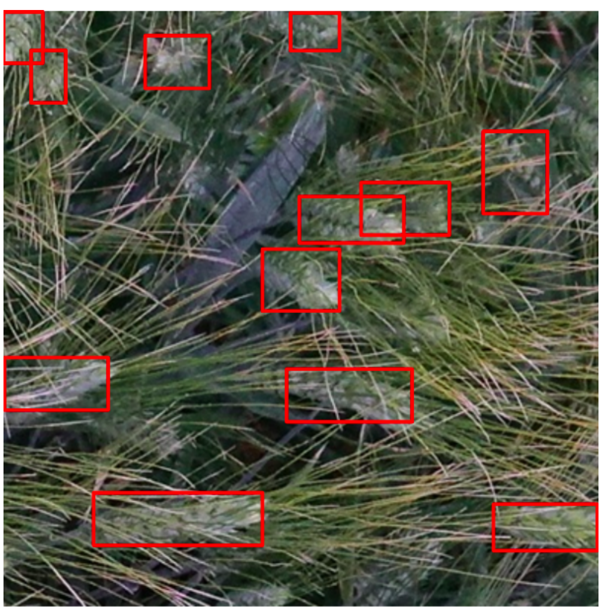


Figure 7. Incorrect predictions by our model. The network is incorrectly predicting plants' yellow green leaves as wheat-head

5.4. Ablation Study

We studied the effect of train and test augmentations on the predictions made by our model. We set all the augmentation probabilities at 0 to test this. The results are in Table 2. We observe about 6% fall in the mAP@0.5 and about 6.4% decrease in WDA values. This is attributed to the dataset's high variance in observation conditions giving rise to diversity in color and shapes. Hence, data augmentation is very important for such small datasets being trained for domain adaptation. This also indicates the dependence of model on data it is trained on which results in less robust model.

Method	mAP@0.5	WDA
Ours	0.635	0.601
Ours (no aug)	0.598	0.562

Table 2. The evaluation results of our model with and without train and test augmentations in terms of WDA and mAP@0.5.

We check the impact of the four fold models trained on different domains of wheat. To check this, we test our model again on test data after removing the first fold model (0th according to index). We calculate the predictions based on the votes of the other three models. These are recorded in Table 3. The difference in WDA is about 3% while 1.5% in mAP@0.5. The reason for little dip is that even after removing one of the trained models, the three other models are still able to help our entire network pipeline generalize well. This decrease in this experiment is lesser than that when data augmentation was removed. This further emphasises the importance of data augmentation.

Method	mAP@0.5	WDA
Ours	0.635	0.601
Ours (without 1st fold model)	0.625	0.584

Table 3. The evaluation results of our model with and without the first (0th) fold pre-trained model in terms of WDA and mAP@0.5.

6. Conclusion

A novel pipeline to improve the results of existing state-of-the-art leading methods is introduced. It uses ensemble technique with semi-supervised pseudo labelling strategy useful for training on unannotated data largely available at low cost. Our method is cost effective as compared to other state-of-the-art methods with lower but comparable results for real time inference. Given a very limited computational budget, the proposed network required only 75% of the cost required by the Faster R-CNN model. Also, our model is able to achieve better mAP@0.5 and WDA results on test images than the baseline YOLOv5s network depicting the potential of our method. Therefore, the implemented method establishes its superiority in terms of its efficiency and effectiveness. With some limitations, the work presented is a significant step in scalability of domain adaptive wheat head detection at little compromise of performance.

Future work includes extending our model for other domain adaptation problems. Our model can also be tested for predicting weeds and pests in the crop by using the corresponding dataset. Future work would entail experiments for a better pipeline which can deal with images with similar RGB characteristics of leaf and wheat head. This method could be used for video of wheat field for real time analysis.

7. Contribution

The authors of this project are Prithvi Chandregowda Venkat and Shreya Chawla. S. Chawla worked on the problem statement selection, proposal writing, researching the topic of object detection and domain adaptation, model design, data analysis, wrote all the code, drafted and gave the project presentation, drafted and reviewed the final report. P.C. Venkat worked on problem statement selection, proposal writing and worked on the report. The authors confirm that P.C. Venkat contributed 20% and S. Chawla contributed about 80% of the entire project.

References

[1] Etienne David, Mario Serouart, Daniel Smith, Simon Madec, Kaaviya Velumani, Shouyang Liu, Xu Wang, Francisco Pinto Espinosa, Shahameh Shafiee, Izzat S. A. Tahir, Hisashi Tsujimoto, Shuhei Nasuda, Bangyou Zheng, Norbert Kichgessner, Helge Aasen, Andreas Hund, Pouria Sadeghi-Tehran, Koichi Nagasawa, Goro Ishikawa, Sébastien Dandrifosse, Alexis Carlier, Benoit Mercatoris, Ken Kuroki, Haozhou Wang, Masanori Ishii, Minhajul A. Badhon, Curtis Pozniak, David Shaner LeBauer, Morten Lilimo, Jesse Poland, Scott C. Chapman, Benoit de Solan, Frédéric Baret, Ian Stavness, and Wei Guo. Global wheat head dataset 2021: an update to improve the benchmarking wheat head localization with more diversity. *CoRR*, abs/2105.07660, 2021. 2, 4, 5

[2] Terrance Devries and Graham W. Taylor. Improved regularization of convolutional neural networks with cutout. *CoRR*, abs/1708.04552, 2017. 2

[3] Glenn Jocher et. al. ultralytics/yolov5: v6.0 - YOLOv5n 'Nano' models, Roboflow integration, TensorFlow export, OpenCV DNN support. `https://github.com/ultralytics/yolov5`, Oct. 2021. 2, 3, 4, 5, 6

[4] Ross B. Girshick. Fast R-CNN. *CoRR*, abs/1504.08083, 2015. 2

[5] Ross B. Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *CoRR*, abs/1311.2524, 2013. 2

[6] Bo Gong, Daji Ergu, Ying Cai, and Bo Ma. A method for wheat head detection based on yolov4, 09 2020. 4

[7] Yoshio Inoue. Satellite and drone-based remote sensing of crops and soils for smart farming – a review. *Soil Science and Plant Nutrition*, 66(6):798–810, 2020. 2

[8] Glenn Jocher. ultralytics/yolov3: v9.5.0 - YOLOv5 v5.0 release compatibility update for YOLOv3. `https://github.com/ultralytics/yolov3`, 2021. 2, 4

[9] Andreas Kamilaris and Francesc Prenafeta Boldú. Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, 147, 04 2018. 1

[10] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. WILDS: A benchmark of in-the-wild distribution shifts. *CoRR*, abs/2012.07421, 2020. 2

[11] Dong-Hyun Lee. Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks. *ICML 2013 Workshop : Challenges in Representation Learning (WREPL)*, 07 2013. 2

[12] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014. 5

[13] Chengxin Liu, Kewei Wang, Hao Lu, and Zhiguo Cao. Dynamic color transform for wheat head detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 1278–1283, October 2021. 2

[14] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. Ssd: Single shot multibox detector. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 21–37, Cham, 2016. Springer International Publishing. 2

[15] Sara Mardanisamani, Farhad Maleki, Sara Hosseinzadeh Kassani, Sajith Rajapaksa, Hema Duddu, Menglu Wang, Steve Shirtliffe, Seungbum Ryu, Anique Josuttes, Ti Zhang, Sally Vail, Curtis Pozniak, Isobel Parkin, Ian Stavness, and Mark Eramian. Crop lodging prediction from uav-acquired images of wheat and canola using a dcnn augmented with handcrafted texture features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019. 2

[16] Hasan MM, Chopin JP, Laga H, and Miklavcic SJ. Detection and analysis of wheat spikes using convolutional neural networks. *Plant Methods*, 2019. 2

[17] Keyhan Najafian, Alireza Ghanbari, Ian Stavness, Lingling Jin, Gholam Hassan Shirdel, and Farhad Maleki. A semi-self-supervised learning approach for wheat head detection using extremely small number of labeled samples. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 1342–1351, October 2021. 2

[18] Michael P. Pound, Jonathan A. Atkinson, Darren M. Wells, Tony P. Pridmore, and Andrew P. French. Deep learning for multi-task plant phenotyping. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct 2017. 2

[19] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *CoRR*, abs/1804.02767, 2018. 2, 4

[20] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region

proposal networks. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. 2, 5

[21] Babak Talebpour, Ufuk Türker, and Uğur Yegül. The role of precision agriculture in the promotion of food security. *International Journal of Agricultural and Food Research*, 4:1–23, 04 2015. 2

[22] Mingxing Tan, Ruoming Pang, and Quoc V. Le. Efficientdet: Scalable and efficient object detection. *CoRR*, abs/1911.09070, 2019. 2

[23] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. *CoRR*, abs/1905.04899, 2019. 2

[24] Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *CoRR*, abs/1710.09412, 2017. 2