Applying High Computed Knowledge Distillation on Pre-trained Model for Edge Computed Grapevine Detection.

HOLLARD Lilian

Université de Reims Champange Ardenne,
Laboratoire LICIIS - LRC CEA DIGIT,
51687 Reims Cedex 2 - France
lilian.hollard@univ-reims.fr

Résumé
The development of Artificial Intelligence has raised interesting opportunities for improved automation in smart agriculture. Smart viticulture is one of the domains that can benefit from Computer-vision tasks through field sustainability. Computer-vision solutions present additional constraints as the amount of data for good training convergence has to be complex enough to cover sufficient features from desired inputs. In this paper, we present a study to implement a grapevine detection improvement for early grapes detection and grape yield prediction whose interest in Champagne and wine companies is undeniable. Earlier yield predictions allow a better market assessment, the harvest work's organization and help decision-making about plant management. Our goal is to carry estimations 5 to 6 weeks before the harvest. Furthermore, the grapevines growing condition and the large amount of data to process for yield estimation require an embedded device to acquire and compute deep learning inference. Thus, the grapes detection model has to be lightweight enough to run on an embedded device. The following of this paper will describe how high-performing computers may help edge computing converge into a state-of-the-art performance. These models were subsequently pre-trained on two different types of datasets and several layer depth of deep learning models to propose a pseudo-labelling Teacher-Student related Knowledge Distillation. Overall solutions proposed an improvement of 7.56%, 6.98, 8.279%, 7.934% and 13.63% for f1 score, precision, recall, mean average precision at 50 and mean average precision 50-95 respectively on BBCH77 phenological stage.

1. Introduction

The improvement of wine quality requires strong control of the vineyards in order to modify the vines by interventions at the plot [18]. Winemakers use yield estimation to get decisive information for the business's economy, plant management, and harvest organization. The interest in knowing the yields early enough involves several economic, administrative, and qualitative objectives [15]. For example, winegrowers can manage the wine market by estimating and controlling the crop volume on a regional or national scale. Also, yield forecasts allow us to know the position of the winegrower's yield in relation to the regulations. For example, the Champagne appellation [10] defines annual production quotas in kg/ha. Finally, yield forecast maps may help to increase the quality of the wines by eliminating part of the harvest by thin-

ning out the bunches, as it has been proven that too high density may hinder the quality of the grapes [23].
Today, most winemakers estimate their yield using manual samplings on selected land plots. It relies on both grape counting and weighting. The counting is visually performed by a human operator, leading to uncertainties on the precision and dubious repeatability. In addition, the weighting of the grapes requires them to be harvested prematurely, a destructive process that cannot be performed extensively. Some work uses historical data to limit grape weighting, significant variations from year to year can skew the predictions. As a result, a variation of 30% can be found between the estimations and the reality [4].
The moment yield forecasts are needed also impacts the solutions. Indeed, most producers would like to obtain estimations at least one month before harvest, which usually falls before the *veraison*, i.e., the start of the maturing period where grapes' colors change. Hence, the color contrast between the leaves of the vine and the green grapes complicates the yield estimation using non-intrusive or non-destructive methods [5]. In addition, the influence of light must be considered for more accurate detection during the day [24].
Artificial Intelligence (AI) bears the promise to improve the work conditions and the accuracy of forecasts as it can better humans in repetitive, time-consuming, and tedious tasks such as counting grapes. Among existing methods, Deep Learning (DL) computer vision is one of the preferred approaches to automatically count grapes in the vineyard [22, 8]. In addition, it shows more robustness and accuracy than other computing vision methods based on signal processing or traditional machine learning (SVM, mostly) [6, 7, 14].
In this work, we focused on the use of accurate and fast deep-learning models for grape counting, the first step towards yield performance forecast. The remainder of the paper is structured as follows : Section 2 reviews existing works in knowledge distillation for grapevine yield estimation and deep learning computer vision. Section 3 presents the main architecture of the proposed solution, the dataset characteristics and the Deep Learning models we have chosen. Section 4 describes the efforts to improve the overall deep learning model performance with Knowledge Distillation. Finally, Section 5 concludes this work.

2. Related Work

Neural networks have been essential to Deep Learning for achieving computer vision state-of-art of object detection. Widely used for their adaptability, generalization and scalability, neural networks showed a correlated performance with the amount of data they train with. Compared to traditional machine learning, deep neural networks has the opportunity to scale better with an increased quantity of data [21].
Among the nowadays commonly used architectures (MobileNet, ResNet, ...) and the detectors that result from them (FasterRCNN, FPN SSD and YOLO, ...), many datasets were born in a joint context of researching the best performance. Datasets such as Microsoft COCO, PASCAL VOC, PlantVillage, and ImageNet consist of a set of standard image data, annotations and commonly joint evaluation procedures. These days, neural network research for object detection papers communicates their results based on these commonly used datasets.
Since Deep Learning neural networks learn, scale and adapt from the features they extract, it has been beneficial to use a pre-trained model as a starting point to train a model with few samples [12]
The remainder of the paper will express the use of a pre-trained model and transfer learning for better training convergence of deep learning model on grapes detection.

In the edge computing community, neural network research has gained a critical focus on optimizing model size, efficiency and computing resources while preserving their accuracy. One concept based on a teacher-student relationship has gained a significant reputation in model compression research. Knowledge distillation refers to using large model predictions to train a smaller model. This process of transferring knowledge from a large and computationally expensive model into a smaller one was successful in several applications of object detection.
In the case of detecting fruits and vegetables for yield estimation application, Thomas A et al. [3] proposed a pseudo-labelling solution using videos. The authors show the overall performance of different deep-learning models for grapes detection using one of the YOLO variants and Mask R-CNN. YOLOv5 architecture experiments on tracking showed the best results in inference computing time and overall detection performance ratio. The pseudo-labelling method improved the mean average precision by +8%.
A. Casado-Garcia et al. [1] conducted a leaf, bunch, pole and wood segmentation for grapevine detection. Automatic yield monitoring with an in-field robotic approach on low-cost cameras for object detection and segmentation solutions requires a complex manual annotation, too time-demanding. The authors proposed three semi-supervised learning methods (Pseudo-labelling, Distillation and Model Distillation) to take advantage of non-annotated images. In their experiments, the authors gained segmentation accuracy between 5.62% and 6.01% on average with semi-supervised learning methods.
Semi-supervised few-shot learning approach for leaf disease recognition also showed significant results with pseudo-labelling techniques. Yang Li et al. [13] boosted the performance of their leaf disease identification model by 2.8% with a single semi-supervised method and by 4.6% with an iterative semi-supervised approach on PlantVillage Dataset. The iterative semi-supervised method describes several training iterations with divided unlabeled samples.
J. Heras et al. [9] showed a +1.78% improvement in semantic segmentation models using pseudo-labelling for grape bunch identification in natural images.
In the research of automated weeding in agrorobotics, Łukasz et al. [2] proposed a lightweight solution running at 10 FPS on Rasberry Pi embedded device with an error reduction of 6.4% with a knowledge distillation technique using the same network structure.

3. Methods and Tools

Our grape detection and counting approach stands on a data collection and real-time analysis technique using Edge computing. The aim is to separate the collaboration between the end-edge and cloud levels by avoiding as much network communication as possible.

3.1. Dataset description

The images used to train the AI model were obtained in a Vranken-Pommery vineyard during the 2022 spring, under natural lighting and weather conditions, covering several weeks preceding the veraison. As the vineyard machinery could not pass during heavy rain or at night, all the images were captured in the early or late morning, with weather conditions ranging from cloudy to clear sky. Different grape varieties were recorded, including Chardonnay, Pinot Noir, and Pinot Meunier.
Because the number of available photos is limited, we also adopted data augmentation methods before the training, allowing us to raise the number of available images.
Also, computer vision is a complex task to compute in an agrorobotics environment without cloud computing relation. Thus, the deep learning model requires special care for an embedded device.

3.2. Deep learning models

Several deep-learning methods can be used to identify an object in an image. Semantic segmentation is a traditional technique to create a precise mapping of the objects in an image, and was applied to grape counting on [16, 17]. However, this technique is computing-expensive and does not scale well for real-time processing.

We focus, therefore, on object detection methods that combine classification and localization. In order to process data streams in real-time, we oriented our research on deep learning algorithms based on a one-step detector. These algorithms use a single neural network to perform object detection and classification with no intermediate tasks. This approach leads to fast models, making them strong candidates for our embedded AI application.

Among the one-step detection models, the YOLO (You Only Look Once) family was selected. The YOLO architecture has been available in several versions since [19]. The original YOLO network is a Deep Fully Convolutional Neural Network that uses a modified GoogleNet as a backbone network, later known as DarkNet-19. Several variants were later published, and YOLOv5 [11] has been significantly optimized to improve detection speed and accuracy. In addition, YOLOv5 is fully compatible with Pytorch and Python, and can be easily set up on edge and mobile devices.

Furthermore, we have made a comparison between YOLO and a two-stage detector called FasterRCNN on phenological stage BBCH77. Composed of region proposal combined with classification head, FasterRCNN from Shaoquing Ren et al. [20] became a very popular object detector for achieving a high average precision with a lower computing time in 2015.

Fully training a complex deep-learning model from scratch is an expensive task. Instead, we applied transfer learning so that an already existing CNN could be adapted to grape detection. In addition, we used the DGX-1 server (Intel® Xeon® CPU ES-2698 v4 @ 2.20 GHz, 8x NVIDIA Tesla V100-SXM2 16Gb GPUs, Python 3+/PyTorch 1.7+) from the ROMEO Supercomputing Center to accelerate the training phase. Furthermore, the supercomputer will serve as a teacher for our edge-developable model, distilling the knowledge of a high computational network into a smaller one. The first concern in transfer learning is to look at the size of the model we want to use for our work. Some models offer complex models and allow high-accuracy performance. Although competitive in terms of accuracy, using large and complex models poses a problem of memory space and size of the neural network.

Therefore, we propose a comparison between the state-of-art of the most undersized object detector with the highest accuracy on the MSCOCO dataset when applying transfer-learning on our dataset of grapes.

TABLE 1 – Mean average precision on BBCH77 grapes detection

	mAP50
Yolov3 Tiny	0.585
Yolov5 Nano	**0.673**
FasterRCNN Resnet50 FPN	0.578

Yolov3 and Yolov5 were the two best models running at least 25 frames per second on our embedded device Jetson Nano 2GB.

As shown in the research community, pre-trained models and transfer learning commonly used to achieve great accuracy with few samples worked well on grapes detection applications. But

can we improve these results with a more specific pre-trained model, trained with grapes detailed features ?

3.3. Pre-trained model and transfer learning

Focusing on the YOLOv5 architecture, we trained our model in two depths of computation. The first one, called Nano, is a miniaturized version and the less computationally expensive version of the YOLOv5 family. The second represents YOLO at its fullest. Called YOLOv5 X, this model require a lot of computing power and is not appropriate for edge computing but expresses the degradation between a large and a compressed model.

TABLE 2 – Performance after transfer-learning on BBCH77 grapes dataset

	F1	P	R	mAP50	mAP50-90
Pre-trained MS COCO - Yolov5 Nano	0.648	0.759	0.565	0.673	0.323
Pre-trained MS COCO - Yolov5 X	0.746	0.865	0.657	0.772	0.456

Among the grapevines dataset, WGISD Dataset stands out by its unmistakable representation of maturated grapes. The pre-trained model will be highly aware of false representative and be a strong candidate for transfer-learning. The model trained on BBCH77 using transfer learning on a WGISD pre-trained model showed a 1.82% overall performance augmentation at the Nano representation.

TABLE 3 – Performance after transfer-learning on BBCH77 grapes dataset

	F1	P	R	mAP50	mAP50-90
Pre-trained WGISD - Yolov5 Nano	**0.66**	0.736	**0.6**	**0.68**	**0.347**
Pre-trained WGISD - Yolov5 X	0.72	0.783	0.668	0.74	0.446

Our main goal is to improve the Nano version of the YOLOv5 architecture to achieve the best performance at the edge. Pre-training the base model with the WGISD Dataset to transfer our dataset afterwards confirms significant results in improving model performance. Using WGISD Dataset helped us to gain 1.8%, 5.83%, 1.02% and 6.91% in f1, recall, mean average precision 50 and mean average precision 50-95.
One interesting fact is the overall downfall of the model X with pre-training on WGISD. Indeed, too much complexity is, in this case, counterproductive. There are not sufficient samples of training data to explore the full potential of the X model. Nevertheless, Recall is slightly better and is one of the most effective component of knowledge distillation.

4. Knowledge-distillation

Knowledge Distillation became a well-known technique for improving small deep-learning models by mimicking more robust and complete neural networks. The most undersized models get more accurate while being suitable for edge and embedded devices. Our paper focus on Response-Based Knowledge. The outputs logits from the Teacher model became a training

sample for the Student model. The main idea is to use the knowledge of a model first trained on true bounding boxes sample and use its acquaintance to fine-tune another model.
Fine-tuning refers to improving a model on the same dataset, like retraining with additional images. This time, the data input remains unchanged, but the bounding boxes are coming from the Teacher model. Implementing early stopping when your model doesn't evolve enough helps define if you are undertraining your model. Undertraining could also explain why your knowledge distillation solutions solve and improve your model. In our case, the Nano model stop by early stopping before achieving the same performance found with knowledge distillation. Applying a teacher-student Response-based relation to solve our problem of grapes detection improved our miniaturized model by 4.34%, 3.41%, 5.06%, 4.89% and 5.96% for f1 score, precision, recall, AP50 and AP50-95.

TABLE 4 – Knowledge distillation improvement for BBCH77 grapes detection

	F1	P	R	mAP50	mAP50-90
BASE - Yolov5 Nano	0.66	0.736	0.6	0.68	0.347
After Knowledge Distillation - Yolov5 Nano	**0.69**	**0.762**	**0.632**	**0.715**	**0.369**

These models were trained on 212 images of grapes between weeks 25 to 26, which correspond to a BBCH77 phenological stage. During our data acquisition Campaign in 2022, we decided to collect as much data as possible even if we knew we could not annotate all the images in the future. Because of this, we have made another group of 183 non-annotated pictures waiting for grape detection training. Since annotating images is a very tedious task and time-consuming, we decided to explore knowledge-distillation techniques and pseudo-labelling methods to improve our models.

4.1. Pseudo-labelling

Since knowledge distillation mainly replaces true target bounding boxes while increasing accuracy, researchers took advantage of it to pseudo-labelled non-annotated images totally unseen by the model. Without modifying any source code of previous knowledge distillation, we can solve the non-annotated targets with Teacher predicted targets.
Applying model X as a pseudo-labeller on the 183 new images boosted our model f1 score by 1.56%.

TABLE 5 – Pseudo-labellisation improvement

	F1	P	R	mAP50	mAP50-90
After Knowledge Distillation - Yolov5 Nano	0.69	0.762	**0.632**	0.715	0.369
After Pseudo-labellisation - Yolov5 Nano	**0.701**	**0.816**	0.616	**0.731**	**0.374**

5. Conclusion

Starting from the base model yolov5 Nano architecture pre-trained on MSCOCO Dataset, we obtained, thanks to 3 layers of optimization, an overall boost of 7.56%, 6.98, 8.279%, 7.934% and

13.63%, respectively for f1 score, precision, recall, ap50 and ap50-90.
In the case of deep learning, pre-trained models are essential to benefit from the already learned feature on a considerable database when having a few examples to train a model with. With this in mind, we kept the idea of fine-tuning a model with an already-seen representation of the targeted object to improve grapevines detection. Knowledge Distillation proves its utility outperforming the base model when it comes to miniaturisation. The model is still improvable by other techniques of distillation. In our study, we only explored Response-based knowledge, but further exploitation is feasible with Feature-based knowledge. Feature-based knowledge captures the distillation loss at every layer of your deep neural network, making it possible to train a student model with the same feature activations as the intermediate layers of the teacher model. Furthermore, if data are insufficient, knowledge distillation helps grapes detection models with non-annotated data enrichment thanks to pseudo-labelling.

Acknowledgment

We want to thank Vranken-Pommery Monopole, our partner in the EDGEAI project, for allowing image collection in their vineyards. We also thank the ROMEO Computing Center [1] of Université de Reims Champagne-Ardenne, whose Nvidia DGX-1 server allowed us to accelerate the training steps and compare several model approaches.

Bibliographie

1. Casado-García (A.), Heras (J.), Milella (A.) et Marani (R.). – Semi-supervised deep learning and low-cost cameras for the semantic segmentation of natural images in viticulture. *Precision Agriculture*, 2022, pp. 1–26.
2. Chechliński (), Siemiątkowska (B.) et Majewski (M.). – A system for weeds and crops identification—reaching over 10 fps on raspberry pi with the usage of mobilenets, densenet and custom modifications. *Sensors*, vol. 19, n17, 2019.
3. Ciarfuglia (T. A.), Motoi (I. M.), Saraceni (L.), Fawakherji (M.), Sanfeliu (A.) et Nardi (D.). – Weakly and semi-supervised detection, segmentation and tracking of table grapes with limited and noisy data. *Computers and Electronics in Agriculture*, vol. 205, 2023, p. 107624.
4. Dami (I.) et Sabbatini (P.). – Crop estimation of grapes. *The Ohio State University Fact Sheet*, 2011.
5. Di Gennaro (S. F.), Toscano (P.), Cinat (P.), Berton (A.) et Matese (A.). – A low-cost and unsupervised image recognition methodology for yield estimation in a vineyard. *Frontiers in plant science*, vol. 10, 2019, p. 559.
6. Diago (M. P.), Tardaguila (J.), Aleixos (N.), Millan (B.), Prats-Montalban (J. M.), Cubero (S.) et Blasco (J.). – Assessment of cluster yield components by image analysis. *Journal of the Science of Food and Agriculture*, vol. 95, n66, 2015, p. 1274–1282.
7. Dunn (G. M.) et Martin (S. R.). – Yield prediction from digital image analysis : A technique with potential for vineyard assessments prior to harvest. *Australian Journal of Grape and Wine Research*, vol. 10, n33, 2004, p. 196–198.
8. Heinrich (K.), Roth (A.), Breithaupt (L.), Möller (B.) et Maresch (J.). – Yield prognosis for the agrarian management of vineyards using deep learning for object counting. *Wirtschaftsinformatik 2019 Proceedings*, 2 2019, p. 15.
9. Heras (J.), Marani (R.) et Milella (A.). – Semi-supervised semantic segmentation for grape

1. https ://romeo.univ-reims.fr

bunch identification in natural images. *In : Precision agriculture'21*, pp. 65–84. – Wageningen Academic Publishers, 2021.
10. INAO. – Aoc champagne - conditions de production.
11. Jocher (G.), Stoken (A.), Borovec (J.), Changyu (L.), Hogan (A.), Diaconu (L.), Ingham (F.), Poznanski (J.), Fang (J.), Yu (L.) et al. – ultralytics/yolov5 : v3. 1-bug fixes and performance improvements. *Version v3*, vol. 1, 2020.
12. Li (X.), Grandvalet (Y.), Davoine (F.), Cheng (J.), Cui (Y.), Zhang (H.), Belongie (S.), Tsai (Y.-H.) et Yang (M.-H.). – Transfer learning in computer vision tasks : Remember where you come from. *Image and Vision Computing*, vol. 93, 2020, p. 103853.
13. Li (Y.) et Chao (X.). – Semi-supervised few-shot learning approach for plant diseases recognition. *Plant Methods*, vol. 17, 2021, pp. 1–10.
14. Liu (S.), Li (X.), Wu (H.), Xin (B.), Tang (J.), Petrie (P. R.) et Whitty (M.). – A robust automated flower estimation system for grape vines. *Biosystems Engineering*, vol. 172, 2018, pp. 110–123.
15. Liu (S.), Marden (S.) et Whitty (M.). – Towards automated yield estimation in viticulture.
16. Mohimont (L.), Roesler (M.), Rondeau (M.), Gaveau (N.), Alin (F.) et Steffenel (L. A.). – Comparison of machine learning and deep learning methods for grape cluster segmentation. – In Boumerdassi (S.), Ghogho (M.) et Renault (É.) (édité par), *Smart and Sustainable Agriculture*, pp. 84–102, Cham, 2021. Springer International Publishing.
17. Mohimont (L.), Steffenel (L. A.), Roesler (M.), Gaveau (N.), Rondeau (M.), Alin (F.), Pierlot (C.), de Oliveira (R. O.) et Coppola (M.). – Ai-driven yield estimation using an autonomous robot for data acquisition. *In : Artificial Intelligence for Digitising Industry Applications*, éd. par Vermesan (O.), John (R.), Luca (C. D.) et Coppola (M.), pp. 279–288. – Rivers Publisher, 2021.
18. Nuske (S.), Achar (S.), Bates (T.), Narasimhan (S.) et Singh (S.). – Yield estimation in vineyards by visual grape detection. – In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2352–2358, 2011.
19. Redmon (J.), Divvala (S.), Girshick (R.) et Farhadi (A.). – You only look once : Unified, real-time object detection. – In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
20. Ren (S.), He (K.), Girshick (R. B.) et Sun (J.). – Faster R-CNN : towards real-time object detection with region proposal networks. *CoRR*, vol. abs/1506.01497, 2015.
21. Sanchez (S.), Romero (H.) et Morales (A.). – A review : Comparison of performance metrics of pretrained models for object detection using the tensorflow framework. – In *IOP Conference Series : Materials Science and Engineering*volume 844, p. 012024. IOP Publishing, 2020.
22. Santos (T.), Bassoi (L.), Oldoni[1] (H.) et Martins (R.). – *Automatic grape bunch detection in vineyards based on affordable 3D phenotyping using a consumer webcam*. – XI Congresso Brasileiro de Agroinformática (SBIAgro 2017), 10 2017.
23. Xi (X.), Zha (Q.), Jiang (A.) et Tian (Y.). – Stimulatory effect of bunch thinning on sugar accumulation and anthocyanin biosynthesis in shenhua grape berry (vitis vinifera × v. labrusca). *Australian Journal of Grape and Wine Research*, vol. 24, n2, 2018, pp. 158–165.
24. Zhang (C.), Ding (H.), Shi (Q.) et Wang (Y.). – Grape cluster real-time detection in complex natural scenes based on yolov5s deep learning network. *Agriculture*, vol. 12, n8, 2022.