Lite CNN Models for Real-Time Post-Harvest Grape Disease Detection

Lucas Mohimont, François Alin, Nathalie Gaveau, Luiz Angelo Steffenel

June 2022

Lite CNN Models for Real-Time Post-Harvest Grape Disease Detection

Lucas Mohimont [a,b,1], François Alin [a], Nathalie Gaveau [b], Luiz Angelo Steffenel [a]
[a] *Université de Reims Champagne Ardenne, LICIIS - LRC CEA DIGIT,*
51687 Reims Cedex 2, France
[b] *Université de Reims Champagne Ardenne, RIBP - EA4707 - USC - INRAE 1488*
51100 Reims Cedex 2, France

Abstract. Post-harvest fruit grading is a necessary step to avoid disease related loss in quality. In this paper, a hierarchical method is proposed to (1) remove the background and (2) detect images that contains grape diseases(botrytis, oidium, acid rot). Satisfying segmentation performances were obtained by the proposed Lite Unet model with 92.9% IoU score and an average speed of 0.16s/image. A pre-trained MobileNet-V2 model obtained 94% F1 score on disease classification. An optimized CNN reached a score of 89% with less than 10 times less parameters. The implementation of both segmentation and classification models on low-powered device would allow for real-time disease detection at the press.

Keywords. classification, segmentation, deep learning, fruit grading, grape disease

1. Introduction

The impact of diseases is one of the main problems in agriculture. Diseases are directly responsible for yield and quality loss in many cultivars. The Food and Agriculture Organization of the United Nation estimates that plant diseases and invasive insects are responsible for 20% to 40% worldwide yield loss [1]. Disease detection is therefore an important problem in the field of Smart Agriculture. Recent advances in computer vision allow for in-field symptoms detection of plant disease. The development of new models is made easier with deep convolutional neural networks that can directly process images. Automatic plant disease detection in field conditions is difficult because there are many challenging factors such as background, natural lighting, symptoms variations, plant phenological stage, etc. Post-harvest detection is therefore still applied during a grading process. Fruits and vegetables grading also includes the detection of others defects such as bruises or lack of maturity.

In this paper, our research is focused on post-harvest grape grading before the pressing stage. Winemakers use preventive pesticide spraying to limit disease proliferation. However, infected grapes can still be harvested if needed. For this reason, a grading process is applied before the pressing to sort the grapes into quality categories. This is necessary to limit the quality loss induced by infected grapes in the press. Our goal is the

[1] Corresponding Author: Lucas Mohimont E-mail: lucas.mohimont@univ-reims.fr

automation of this process with computer vision. Our approach is based on deep learning with semantic segmentation models for automatic grape disease detection. Our two main contributions are the development of an optimized UNet model for automatic grape detection in the existing industrial context and the development of an optimized CNN for disease classification.

The paper is organized as follow: related works are presented in Section 2, the image acquisition protocol, the datasets and the proposed methods are presented in Section 3; then the segmentation and classification results are presented in Section 4, before presenting our conclusions and future works.

2. Related Works

Automated vine disease detection is a dynamic sub-field of AI research in agriculture because it would allow for large scale detection of diseases on the field or after harvest. This work is mainly performed by human operators, whose detection of grape diseases are often biased [2]. In comparison, computer vision over-performs humans in a single grape leaf disease classification task [3], all while ensuring an unbiased evaluation.

Plant disease detection from RGB images is a difficult task. Many difficulties have been described in details in the work of Barbedo [4]. Building a representative dataset is a complex task because there are many sources of variations (phenological stages, disease severity, weather and lighting conditions, etc.). This also leads to heavily unbalanced classes because the symptoms represent a small part of the images.

A first application is grape disease detection from images of individual leaves. This is a popular example because it is easy to develop. The processing pipeline is always the same: (1) images of diseased grape leaves are collected, (2) background removal can be easily done with thresholding segmentation, (3) symptoms regions candidates are detected and then a classifier is applied on every region after feature extraction. Different diseases and methods are proposed in many research works but the environment and task is almost always the same.

Most notably, deep learning with pre-trained GoogleNet model reached 99.35% accuracy on the PlantVillage dataset that contains 14 crop species and 26 diseases. This includes grape leaves affected by Leaf Blight, Black Rot or Esca disease [5]. Deep learning have been recently applied to symptoms quantification with an object detection model based on Faster R-CNN and it reached 81% mean Average-Precision [6].

Practical in-field disease detection is a more difficult task for multiple reasons. The images contain more complex scenes, whole vines instead of a single leaf for example, with natural background and lighting. Multiple deep learning-based models have been proposed recently. Popular object detection architectures, Faster R-CNN [7], R-FCN [8] and SSD [9], were successfully applied on tomato symptoms detection on greenhouse images. A single model reached 83% mean Average Precision on leaf and gray mold, canker, plague, miner, low temperature symptoms, mildew, whitefly pest and nutritional excess [10]. This type of model can be optimized for in-field real time prediction, which would allow for efficient large scale processing.

Data augmentation is systematically used to train deep learning models because they need huge quantity of labeled data. This requires a lot of human time for labeling and computing power for model training. A more classical, feature extraction based, approach does not have this limitation. Such approach was recently applied to in-field downy mildew detection by using color and texture(with the Local Structure Tensor) features with Gaussian mixture modeling and seed-growth correction [11]. A recall rate of 76% with 83% precision rate were obtained on pixel-wise mildew classification from in-field images taken with a flash. These performances are calculated on the labeled pixels and does not entirely represent the performances on mildew spot detection. The labeled dataset was published as a benchmark [12].

Ideally, winemakers want to avoid putting diseased grapes in the pressing batches. In practice, diseased grapes incidence may be too high and could represent a big proportion of the yield. Therefore, post-harvest disease quantification is necessary for grapes grading: healthy grapes and diseased grapes are separated to keep a good wine quality. There are few published research work about post-harvest grape grading from images. The authors of Vazquez-Fernandez proposed a segmentation algorithm based on Gabor filters and pixel classification from raw pixels neighborhoods for grape batch segmentation (ripen grapes, green grapes, rotten grapes, leaves, etc) [13]. A classification success rate of 94% was obtained with multi-layer perceptron at the pixel level. The proposed method used artificial lighting with a simple industrial background to avoid confusion. It does not take the low probability of diseased grapes into account during the prediction, this can result in many false positives. Another published work proposed an automatic stem and leaves detection method from hyper-spectral images [14].

Our two contributions are: (1) grape detection at the press without interfering in the current grading process and (2) the detection of grape crates that contain grapes infected by botrytis, oidium and acid rot. This method would ideally be applied directly in the pressing site without specific calibration and without costly changes in the process.

3. Materials and Methods

The presented work has been carried on three image collection campaigns from 2019 to 2021. This section presents the developed acquisition protocol used in 2019, 2020 and 2021, the collected datasets, the proposed methods and the evaluation metrics.

3.1. Image acquisition

An image acquisition protocol was defined in summer 2019 to collect images for supervised learning. This protocol was designed to be applied directly on the existing industrial setting without too much interference. Data collection was performed in Vranken-Pommery Monopole's wine presses during the weighting stage. The existing industrial process is the following: (1) pallet's containing one to four levels of four grape crates are delivered on the press site; (2) each pallet is moved to the weighting site to collect relevant information (weight, variety, parcels); (3) a quality grade is attributed to the whole pallet after a quick visual inspection by an employee (potential alcoholic content is also estimated during this stage); (4) the pallet is then moved to the docks before pressing.

The quality grade depends on multiple factors like the presence of diseases (oïdium, botrytis, mildew), leaves, soil, etc. It is assumed that the visible area on the upper level of the pallet is representative of the whole pallet. The visual inspection is subjective to the human employee who can be biased by fatigue, visual impairment, etc. Automating this inspection is the goal of the project. GoPro Hero Black 7 cameras, with 4000x3000 resolution, were attached over the weighting sites to obtain images of the visible area of the pallets. The camera's WiFi connection is used with a computer to associate the images to the properties of the pallets (quality grade, variety, weight, alcoholic content). The installation was performed by our team and the data collection was performed by Vranken-Pommery's employees.

Data collection campaigns were carried on in two wine press in 2019 and 2020 in Tours-sur-Marne and Merrey-sur-Arce in France. A third site in Saudoy was added to the campaign in 2021. The industrial process was not changed. The two first sites are in semi-open area and can be affected by the sun light and multiple pallets can be visible in the images. Artificial lighting was only used during the 2021 campaign in Tours-sur-Marne. The resulting datasets were cleaned, few images were removed (redundant takes or dark images taken near sun fall). Each visible crate of the weighted pallet is labeled for binary segmentation with LabelMe polygonal mask [15]. Most images contain 4 crates. Therefore, the segmentation allow us to process each crate separately to detect the disease. Each crates were extracted from the images and they were split into two classes: class 0 for the healthy crates(beside sunburn damages) and class 1 for the crates that contain infected grapes (botrytis, oidium, acid rot).

3.2. Methods

3.2.1. Segmentation for grape detection

Classical image processing techniques can be used for background removal. In this context, thresholding in the HSV space mixed with Gabor-filter-based segmentation, as proposed by [13], was developed and tested on the 2019 dataset. This resulted in about 50% rate of accurate segmentation. This was due to variable lighting, variable grape size in pixels and variable color. Development time was therefore shifted to image labeling and deep learning for image segmentation without subjective choices of features and decision rules. Model training and inference were performed on a Nvidia DGX-1 server with a single Tesla V100 GPU for acceleration.

The proposed segmentation model is a lite Unet model. The Unet [16] model is a popular encoder-decoder neural network designed for semantic segmentation. The encoder is used for feature extraction and the decoder is used for dense pixel prediction. Pre-trained Unet models are available, most of them contains too many parameters. In our case, the Unet model was optimized to allow for real time segmentation. A simplified representation of the proposed model is shown in Figure 1

The encoder is defined as a succession of four pooling blocks. Each block contains two convolutional layers, with a skip connection [17], and a 2x2 pooling layer. Batch normalization was applied before Relu activation for the first layer and without activation for the second layer. The encoder is followed by a similar central block without the pooling layer. Then, the decoder contains one up-sampling block for each block of the encoder. It is similar to the down-sampling block with reversed order (up-sampling-convolution-convolution). There is a skip connection between each pair of down/up-sampling blocks, outputs of both block are concatenated. The segmentation mask is produced by a convolutional layer with a single filter with sigmoid activation.

The size of the model can be changed by increasing/decreasing the number of filters for each layer. In our case, the model should be as small as possible to allow for real-time

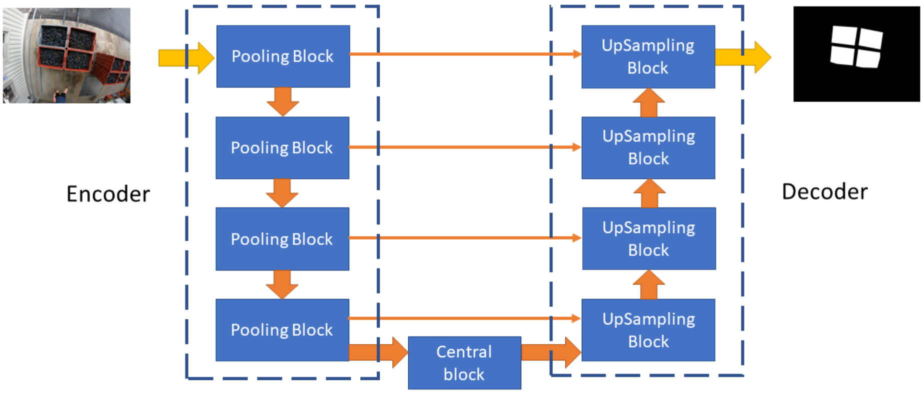

Figure 1. Architecture of the proposed lite UNet model.

prediction during the deployment stage. The number of filters is kept small, between 8 and 64, to limit the model size to 173k parameters. Furthermore, the model is directly applied on the whole images. The image size was reduced from 4000x3000 to 800x600 pixels to fit into memory.

The training was performed on images from 2019, 2020 and 2021. Data-sets were split into training/validation sets for each year, each location and for each color. In total, 1643 images were used for training (970 of red grapes and 673 of white grapes) and 714 were used for validation (421 red grapes, 293 white grapes). Jaccard index and Intersection over Union were used respectively as loss function and segmentation metric. Adam optimizer with a learning rate of 0.0001 and batch size of 16 was used for the training. Early stopping with a patience of 30 epochs was used as regularization.

3.2.2. Classification

The next step is disease detection at the crate level after background removal. The images from the three campaign(2019,2020 and 2021) enabled us to create a large dataset with 3300 images of crates in the ”Healthy” class and 1600 in the ”Disease” class (about 32.6%). Three models were compared after 5-fold cross-validation on 256x256 images. Images were pre-processed with Contrast Limited Adaptive Histogram Equalization(CLAHE) to enhance the contrast [18].

The first model is the MobileNet-V2 [19] classification model pre-trained on the ImageNet dataset. MobileNet models were designed specifically for mobile applications. They serve as backbone for classification, segmentation and object detection architectures. MobileNet-V2 uses depth-wise separable convolutions to reduce the number of parameters. The pre-trained MobileNet-V2 was automatically downloaded by the Keras Applications API. The last layer of the model was replaced by a dense layer with 2 neurons and softmax activation. Fine-tuning was performed with the Adam optimizer with early stopping and with automatic decrease of the learning rate after few epochs of stagnation. Batches of 32 images for a maximum of 200 epochs were used. Training was performed directly on the unbalanced dataset.

The second model is a custom CNN designed to have a small number of parameters. This CNN uses five down-sampling blocks and two dense layers. Each down-sampling block contains two convolutional layers followed by batch-normalization. The first layer

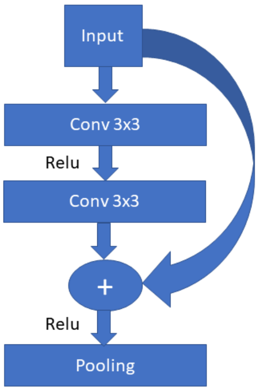

Figure 2. Down-sampling block used in the proposed CNN.

is also followed by a Relu activation. Skip connection and Relu activation is then applied by adding the input and the output of the second layer. Finally, max-pooling is applied to reduce the spatial size of the feature maps. The five blocks uses respectively 8, 16, 32, 64 and 64 filters for each convolutional layers. The block structure is shown in Figure 2. A fully connected layer with 128 neurons was then applied with 50% drop-out probability, the output layer is a dense layer with 2 neurons and softmax activation. The same training configuration was used to train this model. However, full oversampling of the minority class was used to balance the dataset. This model contains only 164k parameters compared to over 2M for MobileNet-V2.

The third model is a baseline classifier based on the classic approach of feature extraction. Six features were computed from the Grey-Level Co-Occurrence(GLCM): dissimilarity, contrast, homogeneity, Angular Second-Moment(ASM), energy and correlation [20,21]. They were computed for each RGB channels. Color features, average value and standard-deviation, were also used. The resulting 24 features were used to trained a SVM classifier with radial-basis function kernel.

Evaluation was performed with the classification metrics: Recall, Precision, F1, area under the ROC curve and area under the Precision-Recall curve.

4. Results

4.1. Segmentation model

The proposed model reached 92.9% IoU on red grapes and 92.6% on white grapes. A segmentation result is shown in Figure 3. Similar performances were observed in every year and every pressing sites. The detailed results for the 2021 model are shown in Table 2. The model was able to detect the grapes in the weighting zone only. In the 2021 validation set, 88 of the 275 images contain visible grapes outside of the weighting zone. However, false positives occurred in only 3 of the 88 images. Multiple errors can be seen in Figure 4: (1) grapes in the background, (2) confusion with the ground and (3) the segmented crates are not completely separated. These kind of errors can be easily removed with morphological operations. Finally, the inference speed of the model was measured

Table 1. Training and validation performances of the models.

Model	% Training IoU	% Validation IoU
UNet-2021 Red Grapes	93.6	92.9
UNet-2022 White Grapes	93.4	92.6

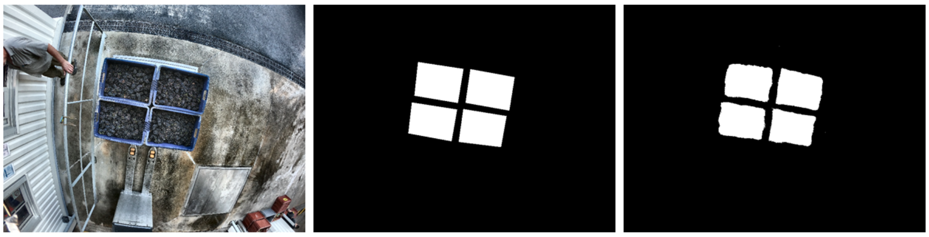

Figure 3. Example of UNet inference: left) Original image from Tours-sur-Marne 2021; center) Labeled mask; right) Predicted mask (93.5% IoU)

Table 2. Intersection over Union for each subgroup of the final dataset.

Split	Training		Validation	
	Red	White	Red	White
Merrey 2019	93.2		92.4	
Tour 2019	91.2	91.5	89	90.2
Merrey 2020	93.3	93	93	90.5
Tour 2020	94	93.4	93.7	92.6
Merrey 2021	91.7	92.3	89.6	93
Tour 2021	93.8	93.8	93.3	93.6
Saudoy 2021	93.5	93.4	93	92.5

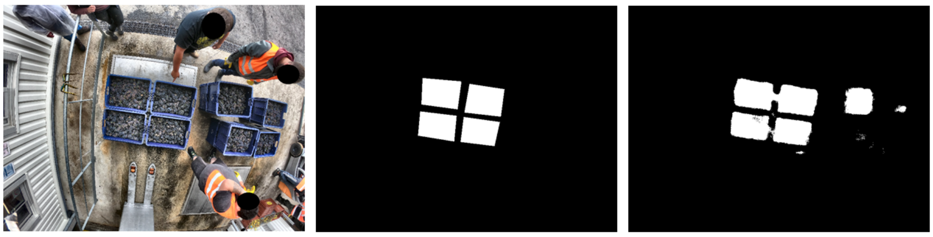

Figure 4. Example of faulty segmentation: left) Original image from Tours-sur-Marne 2021; center) Labeled mask; right) Predicted mask with multiple errors

on the 275 validation images from 2021. The average segmentation time was 0.16s per image. The reduced size of the images and the model, with only 137k parameters, will make optimization on low-powered devices easier.

4.2. Classification model

Results for the Classification model are shown Table 3. The pre-trained MobileNetV2 reached a F1 score of 92%, compared to 89% for the customized lite CNN. MobileNet-V2 fine-tuning was also tested in 2020 and it was unable to converge. We can also note

Table 3. Classification performances of the three models.

Models	Split	Recall	Precision	F1	AUC ROC	AUC PR
MobileNetV2	Training	0.99	0.9	0.94	0.99	0.98
	Validation	0.94	0.9	0.92	0.97	0.95
Lite CNN	Training	0.99	0.94	0.97	0.99	0.99
	Validation	0.93	0.86	0.89	0.97	0.94
GLCM+SVM	Training	0.91	0.87	0.89	0.94	0.92
	Validation	0.82	0.75	0.76	0.92	0.83

that both model obtained similar Precision-Recall AUC with 0.95 for the MobileNetV2 model and 0.94 for the lite CNN. This is important because the lite CNN has only 164k parameters compared to 2.2M parameters for MobileNetV2. The lite CNN could be easily implemented on a low-power device with a smaller classification threshold to allow for real-time prediction directly in the pressing site. We also note that the gap between training and validation scores is bigger for the lite CNN compared to the MobileNet model. This is probably due to duplication oversampling. Training with data augmentation could be necessary to reduce over-fitting.

This gap was even bigger for the GLCM+SVM model with 89% training F1 and 76% validation F1. This is mostly due to the precision rate of 75%. In this case, deep learning was able to surpass the classical method with a difference of +10% in recall, precision and F1 scores. This does not mean that a classical method could not reach the same level of accuracy. But it would imply extensive development for feature extraction, features and models selection. The choice might depend on the hardware available to the researchers. Deep learning for images needs at least one powerful, expensive, GPGPU. Classical methods can be executed without GPGPU (they still need careful optimization) but might need more development time. Classical methods are also easier to interpret because every steps are defined by the developer. In the case of deep learning, neural networks have the reputation to act like black boxes. The weights of a neural network can not be directly interpreted, unlike logistic regression for example. Visualization tools have been proposed for this purpose. Algorithm like t-SNE and Umap [22] are used to project neural networks activations(or feature maps) into 2-D or 3-D data that can be visualized. The Umap algorithm was applied to our case. We compared the pre-trained MobileNet before and after fine-tuning to show the evolution of the discriminative capacity induced by the training process. The algorithm was applied with 100 neighbors with a minimal correlation distance of 0.9. We can see in Figure 5 that the two classes are not completely separated when using the pre-trained ImageNet features. The separation is more distinct after fine-tuning.

5. Conclusion and Future Works

The proposed lite models for segmentation and classification reached respectively 0.92% IoU and 0.89% F1 scores with less than 200k parameters. In particular, the lite CNN classification model has 10 times less parameters compared to a pre-trained MobileNet-V2 model with 0.94% F1 score. A compromise between accuracy and inference speed is necessary for a practical implementation of the existing models for real-time usage. A previous segmentation model was already optimized for embedded execution in STM32

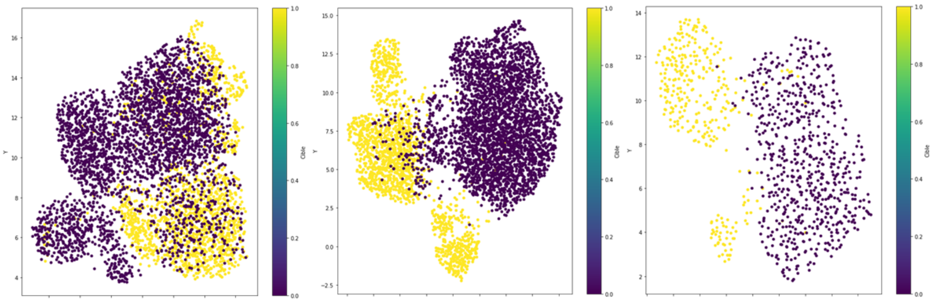

Figure 5. Umap applied to the MobileNetV2 models. Points in yellow are positives samples(disease) and points in purple are negatives samples(healthy): left) Pre-trained MobileNetV2 before fine-tuning; center) MobileNetV2 after fine-tuning; right) MobileNetV2 after fine-tuning, validation set only.

boards and Jetson Nano [23]. Better segmentation speed is expected with the current model. Future works also includes the development of segmentation models for disease detection and quantification. Implementation and evaluation of these models are expected for the next campaign in late summer 2022.

Acknowledgements

This work has been performed in the project AI4DI: Artificial Intelligence for Digitizing Industry, under grant agreement No 826060. The project is co-funded by grants from Germany, Austria, Finland, France, Norway, Latvia, Belgium, Italy, Switzerland, and the Czech Republic and - Electronic Component Systems for European Leadership Joint Undertaking (ECSEL JU).

We want to thank Vranken-Pommery Monopole, our partner in the AI4DI project, for allowing image collection in their sites. We also thank the ROMEO Computing Center[2] of Université de Reims Champagne-Ardenne, whose Nvidia DGX-1 server allowed us to accelerate the training steps and compare several model approaches.

References

[1] Secretariat I, Gullino M, Albajes R, Al-Jboory I, Angelotti F, Chakraborty S, et al. Scientific review of the impact of climate change on plant pests. FAO on behalf of the IPPC Secretariat; 2021. Available from: DOI:10.4060/cb4769en.

[2] Hill GN, Evans KJ, Beresford RM, Dambergs RG. Comparison of methods for the quantification of botrytis bunch rot in white wine grapes. Australian Journal of Grape and Wine Research. 2014;20(3):432-41. Available from: DOI:10.1111/ajgw.12101.

[3] Cruz A, Ampatzidis Y, Pierro R, Materazzi A, Panattoni A, De Bellis L, et al. Detection of grapevine yellows symptoms in Vitis vinifera L. with artificial intelligence. Computers and Electronics in Agriculture. 2019;157:63-76. Available from: DOI:10.1016/j.compag.2018.12.028.

[4] Barbedo J. Factors influencing the use of deep learning for plant disease recognition. Biosystems Engineering. 2018 08;172. Available from: DOI:10.1016/j.biosystemseng.2018.05.013.

[5] Mohanty SP, Hughes DP, Salathé M. Using Deep Learning for Image-Based Plant Disease Detection. Frontiers in Plant Science. 2016;7. Available from: https://www.frontiersin.org/article/10.3389/fpls.2016.01419.

[2] https://romeo.univ-reims.fr

[6] Xie X, Ma Y, Liu B, He J, Li S, Wang H. A Deep-Learning-Based Real-Time Detector for Grape Leaf Diseases Using Improved Convolutional Neural Networks. Frontiers in Plant Science. 2020;11. Available from: `DOI:10.3389/fpls.2020.00751`.

[7] Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks; 2016.

[8] Dai J, Li Y, He K, Sun J. R-FCN: Object Detection via Region-Based Fully Convolutional Networks. In: Proceedings of the 30th International Conference on Neural Information Processing Systems. NIPS'16. Red Hook, NY, USA: Curran Associates Inc.; 2016. p. 379–387.

[9] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, et al. SSD: Single Shot MultiBox Detector. Lecture Notes in Computer Science. 2016:21–37. Available from: `DOI:10.1007/978-3-319-46448-0_2`.

[10] Fuentes A, Yoon S, Kim SC, Park DS. A Robust Deep-Learning-Based Detector for Real-Time Tomato Plant Diseases and Pests Recognition. Sensors. 2017;17(9). Available from: `https://www.mdpi.com/1424-8220/17/9/2022`.

[11] Abdelghafour F, Keresztes B, Germain C, Da Costa JP. In Field Detection of Downy Mildew Symptoms with Proximal Colour Imaging. Sensors. 2020;20(16). Available from: `DOI:10.3390/s20164380`.

[12] Abdelghafour F, Keresztes B, Deshayes A, Germain C, Da Costa JP. An annotated image dataset of downy mildew symptoms on Merlot grape variety. Data in Brief. 2021;37:107250. Available from: `DOI:10.1016/j.dib.2021.107250`.

[13] Vazquez-Fernandez E, Dacal-Nieto A, Martin F, Formella A, Torres-Guijarro S, Gonzalez-Jorge H. A computer vision system for visual grape grading in wine cellars. In: International Conference on Computer Vision Systems. Springer; 2009. p. 335-44.

[14] Portalés C, Ribes-Gómez E. An image-based system to preliminary assess the quality of grape harvest batches on arrival at the winery. Computers in Industry. 2015;68:105-15. Available from: `DOI:10.1016/j.compind.2014.12.010`.

[15] Wada K. labelme: Image Polygonal Annotation with Python; 2016. `https://github.com/wkentaro/labelme`.

[16] Ronneberger O, Fischer P, Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Navab N, Hornegger J, Wells WM, Frangi AF, editors. Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. Cham: Springer International Publishing; 2015. p. 234-41.

[17] He K, Zhang X, Ren S, Sun J. Deep Residual Learning for Image Recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 2016. p. 770-8.

[18] Pizer SM, Amburn EP, Austin JD, Cromartie R, Geselowitz A, Greer T, et al. Adaptive histogram equalization and its variations. Computer Vision, Graphics, and Image Processing. 1987;39(3):355-68. Available from: `DOI:10.1016/S0734-189X(87)80186-X`.

[19] Sandler M, Howard A, Zhu M, Zhmoginov A, Chen LC. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In: IEEE Confe. on Computer Vision and Pattern Recognition (CVPR); 2018. .

[20] Hall-Beyer M. GLCM Texture: A Tutorial v. 3.0 March 2017. PRISM; 2017. Available from: `DOI:10.11575/PRISM/10182`.

[21] Haralick RM, Shanmugam K, Dinstein I. Textural Features for Image Classification. IEEE Transactions on Systems, Man, and Cybernetics. 1973;SMC-3(6):610-21.

[22] McInnes L, Healy J, Melville J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv; 2018. Available from: `https://arxiv.org/abs/1802.03426`.

[23] Roesler M, Mohimont L, Alin F, Gaveau N, Steffenel LA. Deploying Deep Neural Networks on Edge Devices for Grape Segmentation. In: Boumerdassi S, Ghogho M, Renault É, editors. Smart and Sustainable Agriculture. Cham: Springer International Publishing; 2021. p. 30-43.