VINBOT: robot vision and learning from vineyards

João Pedro Mak Duarte

Thesis to obtain the Master of Science Degree in

Electrical and Computer Engineering

Supervisors: Prof. José Alberto Rosado dos Santos Victor
Prof. Jorge dos Santos Salvador Marques

Examination Committee

Chairperson: Prof. João Fernando Cardoso Silva Sequeira
Supervisor: Prof. José Alberto Rosado dos Santos Victor
Member of the Committee: Prof. Ricardo Nuno da Fonseca Garcia Pereira Braga

October 2020

Declaration

I declare that this document is an original work of my own authorship and that it fulfils all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Acknowledgments

Nenhuma viagem se pode cingir ao seu fim. Em primeiro lugar, gostaria de agradecer aos meus orientadores, o Prof. José Santos-Victor e o Prof. Jorge Salvador Marques, pelo aconselhamento e apoio prestado ao longo de todo o processo de desenvolvimento desta tese. Agradeço também à Dra. Catarina Barata pela ajuda e disponibilidade. Ao Eng. Gonçalo Victorino, o meu mais profundo obrigado, pela camaradagem, amizade e pelo, aparente, inesgotável tempo para me responder a todas as perguntas, sempre com empenho e zelando pelo meu interesse.

Em segundo lugar, gostaria de agradecer à minha família, em todo o espectro etimológico da palavra. Um obrigado aos meus avós, obrigado pelo amor incondicional, estarão para sempre num patamar diferente dos outros. À minha mãe, agradeço ter sempre feito o melhor que podia comigo, ter-me posto em 1º lugar e ter-me mostrado mundos novos. Ao meu pai agradeço tudo. Todas as suas ações são comigo em mente, com o que for melhor para mim, por muito que isso lhe possa custar. Obrigado por me ouvires, sempre. Ao meu irmão, por ser quem é, agradeço. Obrigado por cuidares sempre de mim. Agradeço, também, à minha tia Beatrice e ao meu tio Carlos, pelas conversas, puxões de orelhas e amor. Obrigado João Diego e Filipa, por serem minha família e pela comida.

Por fim, queria agradecer aos meus amigos, aqueles que são, também, família. Obrigado Teresa, Tiago, Miguel e Sandra, pelas lições, verbais ou empíricas e por existirem na minha vida. Quero agradecer aos meus amigos que a faculdade me trouxe, em especial ao grupo de trabalho SIIIIIIIIIM, sem o qual não estaria onde estou. Um grande e sentido obrigado ao Frederico, por seres meu amigo há 18 anos, por tudo aquilo que já passámos. Ritinha, pelos abraços, por me ensinares tanto, por me tornares uma melhor pessoa e, principalmente, por seres quem és, obrigado. À table rouge, agradeço a presença nos meus anos formativos.

Obrigado a todas as pessoas que contribuíram para a incrível pessoa que hoje sou.

Abstract

The goal of this work is to estimate a vineyard's yield based on the visible area of grape bunches resulting from an autonomous segmentation in a set of images of a vineyard.

Firstly the problem of autonomous segmentation is tackled by the use of a Fully Convolutional Network (FCN), trained with data from the Instituto Superior de Agronomia (ISA) vineyard with minor data augmentation, operation which increases the number of images. The FCN is tested and its loss function adjusted to compensate the imbalance present in the data set, where the grape clusters only represent 3.5% of the entire images. This is complemented by pre and post processing operations that improve the segmentation's score, the Intersection Over Union (IOU). This metric evaluates how well the segmentation overlaps the ground truth. The pre processing is composed by a sliding window and a colour space change that increased the test set score to 62%. As for the post processing, the morphological operation "open" is used and the image rebuilt with the objective of removing false positives. The combination of these efforts result in a IOU score of 64%.

In the second part of testing, the yield is estimated with the use of two models, one that predicts the percentage of grape bunches hidden in the image according to the porosity of the vine, and another that transforms the total area of bunches into volume. Four different cases are presented, two varieties, encruzado and arinto, from the same to stages, harvest and veraison. The veraison results achieve the desired metric score of an error less than 10% for both varieties, 3% for encruzado and just under 10% for arinto. Although some aspects of the overall process need improvement, in order to make it more robust, the results were satisfactory for this part.

Keywords

Computer vision, Precision Viticulture, Yield estimation, Machine learning

Resumo

O objectivo principal deste trabalho é de estimar atempadamente o peso total das uvas após a vindima, com base numa segmentação autónoma de cachos de uvas em imagens de vinhas.

Primeiramente, o problema da segmentação é abordado pelo uso de uma FCN, treinada com dados recolhidos na vinha do ISA, que por sua vez foram multiplicados com operações de desdobramento de dados. A FCN foi testada e a função de custo foi ajustada de forma a que representasse o desequilíbrio que existe entre classes nas imagens. Esta mudança é também complementada com pré e pós-processamento que melhoram o resultado da métrica de avaliação, o IOU, que avalia quão bem a segmentação efectuada se sobrepõe à realidade. O pré-processamento é composto por uma janela deslizante sobre a imagem e uma mudança de espaço de cor que melhorou a métrica para 62% no conjunto de teste. No pós-processamento, a operaçao morfológica ”open” é usada e a imagem é reconstruída, ambos com o objectivo de remover falsos positivos. O resultado atingido é de 64% IOU.

A segunda parte de testes é direcionada para a estimativa de peso. São usados dois modelos, um que estima a percentagem de cachos escondidos numa imagem por folhas ou outros cachos, através da porosidade da vinha, outro que transforma área em peso. São usados quatro cenários diferentes, duas castas, arinto e encruzado, em duas fases, pintor e fim de maturação. Os resultados da fase pintor atingem o objectivo de menos de 10% de erro relativo ao real, com 3% de erro na casta encruzado e quase 10% na casta arinto. Embora o processo precise de melhorias nalguns aspectos de forma a aumentar a robustez de todo o sistema, este conseguiu resultados satisfatórios.

Palavras Chave

Visão computacional, Viticultura de precisão, Estimativa de peso, Aprendizagem automática

Contents

List of Figures

List of Tables

Acronyms

ACC	Accuracy
CNN	Convolutional Neural Networks
CIE LAB	Commission internationale de l’éclairage Lightness* a* b*
FCN	Fully Convolutional Network
F-RCNN	Region-Based Convolutional Neural Networks
GRVI	Green–Red Vegetation Index
HOG	Histogram Oriented Gradient
HSV	Hue Saturation Value
IOU	Intersection Over Union
ISA	Instituto Superior de Agronomia
LBP	Local Binary Pattern
ML	Machine Learning
NN	Neural Networks
PA	Precision Agriculture
ReLU	Rectified Linear Unit
RGB	Red-Green-Blue
SVM	Support Vector Machine
SP	Smart Points

1

Introduction

Contents

1.1 Motivation

Since automation of systems began to be a standard in every area of human production, it has increased crop output in agriculture, reduced manual labour and created an overall improvement in quality of life, according to [10]. Also, the more frequent use of robotics in agriculture is due to the lack of human resources relative to manual labour and to the increasing business competitiveness.

These developments include harvesting, seeding, irrigation and other type of robots related to the basics of agriculture activity. Along with the new possibilities that technology brings, also new strategies have been developed in relation to agriculture, one of which being Precision Agriculture (PA). PA has been the trend in most crops. The idea is that each parcel of the field is different, and as such, should have different needs. As an example, inside a vineyard, vines located on a hill have less access to water than the ones directly next to them, on a plane, needing more irrigation. With PA, new challenges are brought to attention. This more detailed information over a field, in this particular case a vineyard, allows for a more precise control over the plantation over more precise techniques related to yield estimation, quality analysis or post harvest production.

The first and foremost issue with yield estimation is the vine's natural variability. Any vine may give significantly different yields depending on the year (temporal variability), soil or weather conditions, biotic or abiotic stresses, variety or agriculture practices [11–13]. Given the difficulties exposed and with the advancements of robotics, especially sensor-based technology, some works have been made in order to be able to develop an automated system of yield estimation along the vine's natural cycle. One of the paths pursued is the use of computer vision. Having images as data, Machine Learning (ML) applied to image processing is one of the most common trends [3, 4, 14, 15].

These activities have taken a prominent importance in the current agriculture research. That being said, the study of an accurate yield estimation system, regardless of the crop in question, has increasingly become a necessity. In the specific case of viticulture, an accurate yield estimation brings significant advantages such as: correct estimation of cellar needs, the possibility of developing targeted marketing strategies, knowing in advance the amount of machinery and manpower needed for harvest, allocating cellar space and equipment and managing stock prices for both the grapes and the produced wine [16].

In ML, with the development of processing units and new open-sourced programming libraries, the difficulty of applying not only classical methods, such as statistical models, but Neural Networks (NN) as well, has decreased.

1.2 Problem formulation

Instituto Superior de Agronomia (ISA) is developing a project with the goal of estimating a vine's yield without invasive operations. This project is based on a moving robot with sensors that collects data along their vineyard's lines. This data is, at the present moment, of two kinds: Red-Green-Blue (RGB) images and porosity information. After collection, the data is used for two purposes, the first being the segmentation of grape clusters that are visible in the image and the second to determine the percentage of clusters that are covered by leaves. This percentage is obtained from the porosity data, given that there is a correlation between the porosity and the amount of non-visible clusters [17]. Knowing the total area of clusters in the image, the next step is to transform that area into weight. This is done through a model also developed at ISA, providing the final yield estimation.

At the moment, no part of this process is automatic. That being said, the main goal of this work is to create an algorithm that has as an output the visible area of grape clusters in any given image provided by the mobile platform, in order to automatise this step of the process. This first step in automation has additional problems other than the main one of creating a system to replace the hand segmentation. The human made steps create a lot a variability in the several stages of this process. For example, the data collection is made in a way that the robot is not always at the same distance to the vine nor it is made at the same time of day, sometimes resulting in images with difficult lighting conditions. The ground truth is also hand made, which may induce error in both the segmentation stage as in the following stages. These problems, combined with the main one of providing automation for a stage of a non-invasive yield estimation model, form the work these thesis aims to solve.

1.3 Outline

In Chapter 2, the main practises in precision viticulture referring to yield estimation will be presented, alongside a few projects that also relate ML to the problem and a brief review of image segmentation techniques.

In Chapter 3, the system that will be used in solving the problem is presented and described in detail, starting with the necessary image pre-processing, passing through the grape bunch segmentation, followed by the prediction post-processing and yield estimation.

In Chapter 4, experiments are made on the Encruzado variety in order to better understand which variables of the system are best for the task and then on a new data set the yield estimation is tested. The results analysed and discussed.

Finally in Chapter 5, tasks and ideas that could improve the overall of this process and provide continuity are proposed and the project's conclusions are shown.

2

Related Work

Contents

In this chapter, some basic notions are presented relative to the viticulture aspect of this work. Also, other works that address in some way the same goal, or the goal of any step, of this project are discussed.

2.1 Yield estimation

As a generalised practise [18], the way to estimate yield requires a deep knowledge of the vineyard variability in space and time, combined with years of expertise in viticulture. Firstly, a set of samples is taken from several parts of the vineyard where the producer knows to be different from one another. Following the sampling, the producer weighs the set and extrapolates for the patch where it was taken from and, finally, the estimates are added, resulting in the final prediction. The way the extrapolation is made varies from producer to producer since it also takes into consideration empirical knowledge. These sort of methods are time consuming and can be destructive to the crops.

For Precision Agriculture (PA), yield estimation has been not only convenient but necessary. For this specific problem, new alternative methods have been developed and used commercially. Usually these methods present some limitations, the main one being that they rely on invasive techniques such as defoliation, as shown in Figure 2.5, or that they are aimed at yield estimation at a larger scale.

Given the utmost importance of being able to estimate the yield , several efforts have been made to develop new strategies and technology that support this area. Some methods are already in use, like the aeropalynological forecast models [19], although this is more directed at a regional scale production estimate. This method is based on vineyard pollen readings and correlates the amount of pollen concentration in the air of a certain region which increases with the number of flowers and, consequently, the number of future grapes. So, the current trend has been in sensor based technology. Another method that is still under development is the one proposed by [20], where is said the tensile strength of the vineyard's supporting wires is adjusted to be proportional to the weight of the existing bunches. This method has the limitation of requiring large investments in sensors that will also require regular maintenance. Although quite a few different approaches have been made, the main trend has been visual-based methods. There are some currently under development and others already tested. They will presented in further detail in the following sections.

2.2 Computer vision in the viticulture and agronomic context

Computer vision has become one of the most common strategies adopted in the yield estimation problem, not only regarding bunch recognition [16, 21], but also shoots [22], flowers [23, 24] and berries [25, 26]. Using computer vision, it is reasonable to assume that a more correct identification of the yield components(bunches, berries, flowers or shoots) will provide a more accurate estimation.

2.2.1 Image Segmentation

Consequently, looking at reviews from the recent years, the main work effort has been oriented towards computer vision, more specifically in Machine Learning (ML) [27–29]. In particular, Neural Networks (NN) have demonstrated to outdo expectations in several fields, in particular the Fully Convolutional Network (FCN), presented in [1], when trained end-to-end, pixels-to-pixels on semantic segmentation exceed the previous best results without further machinery. Another approach to this problem is the technique of transfer learning. Having the problem of lack of data, it is possible to adapt a existing pre-trained classifying NN, changing the fully connected layers into a deconvolution, as proposed by [1]. This utilises the feature learning part of the network, being that the only trained part is the deconvolution or upsampling part, with the data specific to the problem(Figure 2.1). This structure is further explained in Chapter 3.

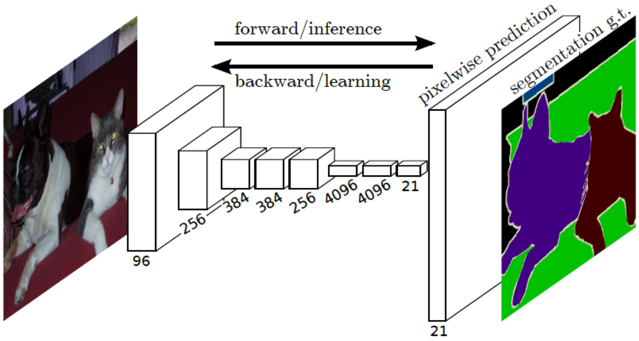


Figure 2.1: FCN architecture
(Source: [1])

Of course the limitations to any NN solution revolve around the lack of direct control over the classification process and feature learning, the possibility of over fitting and the inability to be certain when a minimum of the loss function is reached, that it is the global minimum that will solve the problem optimally. Also, the amount of data that is usually required is significantly more than the amount used for traditional image segmentation methods.

Another important work done in image segmentation based on Convolutional Neural Networks (CNN) is [2]. The U-Net (Figure 2.2) proposed, is comprised of two stages: firstly the compression stage that is dedicated to feature extracting resulting in a multi-channel feature map; and in the second stage, it is added a usual contracting network by successive layers, replacing the pooling layers with upsampling operators, increasing the output's resolution. This network provides the possibility of end-to-end training with a reduced data set, obtaining satisfying results.

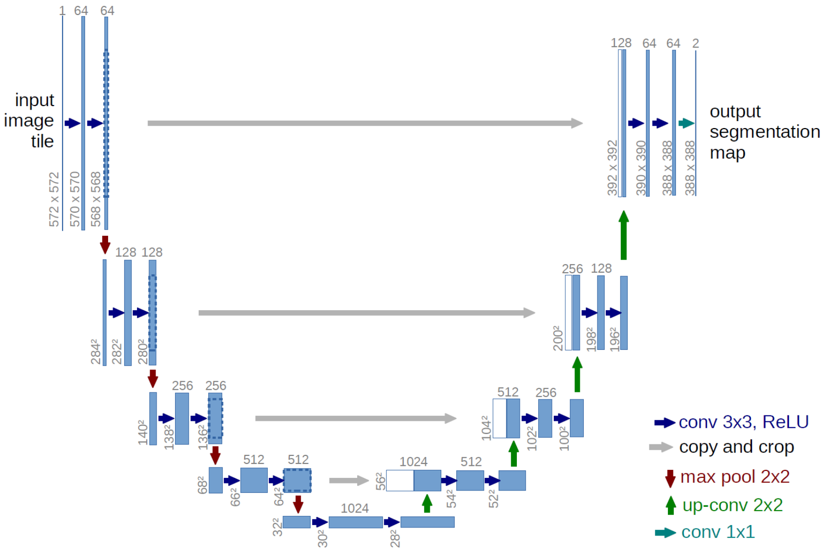


Figure 2.2: U-Net architecture
(Source: [2])

Regardless, to opt for a classic approach in segmentation through computer vision would also be an option, with the advantages of having a greater control of the overall process, resulting in an easier feature tweaking, on one hand, but on another this process may be more time consuming and difficult. That being said, some works combine the two approaches in separate steps of the process, like [4, 30], the first using a combination of FCN with the application of a Hough Transform based method and the second using descriptor vectors that combine Histogram Oriented Gradient (HOG) and Local Binary Pattern (LBP) followed by a Support Vector Machine (SVM). Another work that combines classical methods with more recent ones, is [31]. With the objective of immature green citrus fruit detection, it performs fruit detection through booth Region-Based Convolutional Neural Networks (F-RCNN) and a multi-level Hough circle method.

2.2.2 Yield components recognition

Taking a NN approach to this problem, there are already projects who tackle the same problematic with a similar backbone idea [3, 4, 14, 31] that show promising results.

One of the most interesting strategies is described in [3]. In this paper transfer learning is applied. A pre-trained Classifying Neural Network, the Inception-V3 [32] is used as a base for the localisation algorithm. The authors choose to use not only the architecture but the already trained weights of this NN since one of their limitations was the scarcity of data, labelled or otherwise. To be able to correctly classify and localise the bunches in the images, the last layer was replaced by what the authors entitled ”localisation head”. This head takes the information from the second to last layer and uses it to produce an outcome of probability of a certain area in the image having or not a bunch. Through this probability map, a bounding box is created around the areas with the largest probability value. This was originally designed for apple orchards. The localising head was trained separately from the remaining network with images labelled with containing apples or not. This fine tuning is what allowed the Network to correctly classify and segment the apples in the images. Later on, the fine tuned NN was trained on images that contained grape bunches. For this later training, the algorithm split the images into areas of interest, (Figure 2.3) and could correctly classify 99% of them as for containing or not grape bunches.

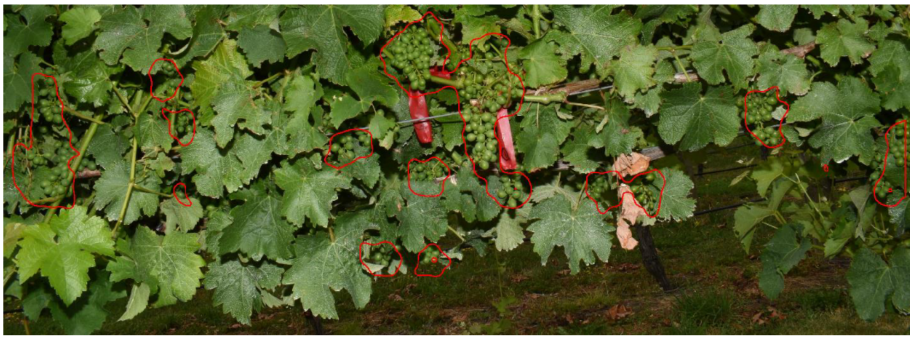


Figure 2.3: Resulting grape bunches bounding polygons
(Source: [3])

In [14], the last layer of the NN is replaced with a classification layer made by five neurons, one for each possible classification, bunch, wood, pole, leaves and background. This last layer can be described as a *maxpool* layer, in which the classification is given through the most likely probability of the patch in analysis. The algorithm was tested with four different NN:

- **Alexnet** [33]: A feed-forward sequence of 5 convolutional layers and 3 fully-connected layers. The first convolutional layer has a size of 11x11, whereas the remaining have filter size of 3x3;
- **VGG16** [34]: A feed-forward sequence of 13 convolutional layers and 3 fully-connected layers. All convolutional layers have filter sizes of 3x3;
- **VGG19** [34]: A feed-forward sequence of 16 convolutional layers and 3 fully-connected layers. All convolutional layers have filter sizes of 3x3;
- **GoogLeNet** [32]: A feed-forward sequence of 3 convolutional layers of 7x7, 1x1 and 3x3 filter sizes, 9 inception layers, each one made of 4 parallel blocks (without convolution and with 1x1, 3x3, 5x5 convolutional layers), and a single fully-connected layer at the end of the chain.

Since all of the networks presented above have input size specification, the input images had to be resized to fit their specific architecture. With the different goal of simply counting the amount of bunches in a given image, this strategy was based on a 80x80 pixel sliding window. The 80x80 size was selected from a mean size of a bunch in an image in their specific data set. This window would be then resized and fed to the NN. The Accuracy (ACC) results for each NN are shown in Table 2.1:

Table 2.1: Accuracy results for the testing NN

NN	ACC(%)
Alexnet	81.03
VGG16	83.05
VGG19	91.52
GoogLeNet	79.66

Like previously mentioned, although this paper's objective [14] is not aligned with the purposed work in this one, it introduces a few ideas that may be particularly useful, such as the Green–Red Vegetation Index (GRVI). This metric is computed as follows in Equation (2.1):

$$GRVI = \frac{\rho_{green} - \rho_{red}}{\rho_{green} + \rho_{red}} \tag{2.1}$$

Where ρ_{green} and ρ_{red} is the reflectance of visible green and red, respectively. In terms of balance between green and red reflectance, three main spectral reflectance patterns of ground cover can be identified:

- green vegetation: where ρ_{green} is higher than ρ_{red};
- soil (e.g., sand, silt, and dry clay): with ρ_{green} lower than ρ_{red};
- water/snow: where ρ_{green} and ρ_{red} are almost the same.

Another approach to the identification of yield components is the one presented in [4]. Their aim is one that is very similar to the work of this dissertation, with the difference that the data collection is performed during the flowering stage of the grape vine. Their process is divided into two stages: localisation of inflorescences in the image and single flower extraction. The first step is done through the use of a FCN, with a encoder part adapted from the AlexNet [33] and a decoder with only two up-convolutions(Figure 2.4). This architecture is derived from the U-Net [2]. They train the network with labelled images of the vines, with the classes of inflorescences and not-inflorescences.

Focusing on the inflorescence detection, since they use a process based on the Hough transform for the flower extraction part, 5292 608x608 pixel images were used to train the network. Given the nature of the problem, the detection and localisation of inflorescences results in regions of interest (ROI) and, as such, mean Intersection Over Union (IOU) was used as a quality measure.

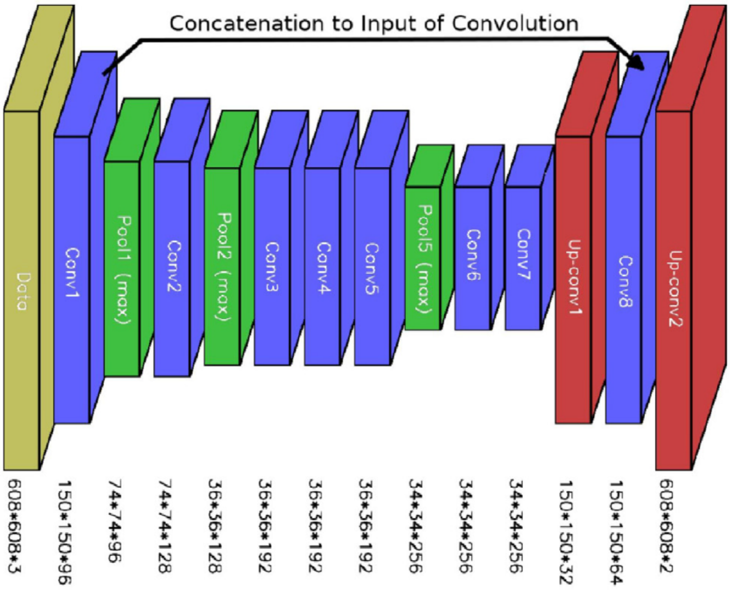


Figure 2.4: [4] Network's detailed architecture
(Source: [4])

The best results were after the 285500^{th} epoch, which resulted in a mean IOU of 87.6%, with a class-specific IOU of 76% for the inflorescence class.

2.2.3 Limitations

The previously mentioned methods, with the exception of [4] lack practicality in some sense. For example, part of them used a data set with vineyards that have had some of their leaves removed, as in Figure 2.5. Also, for the specific goal of this work, that is estimating yield, when recognising bunches in an image, the objective is to segment them in the image, not only count them, as it is done in [14]. The main limitation, in general, is that to base a method of yield prediction on the bunches that exist, the more of them that are occluded, either by other bunches or leaves, the more uncertain the method is going to be. This work aims to overcome these limitations, starting with not being invasive. The data collection does not interfere with the natural vine development or viticulture practises. Also, with the prediction models developed at Instituto Superior de Agronomia (ISA), the occlusion problem is handled.

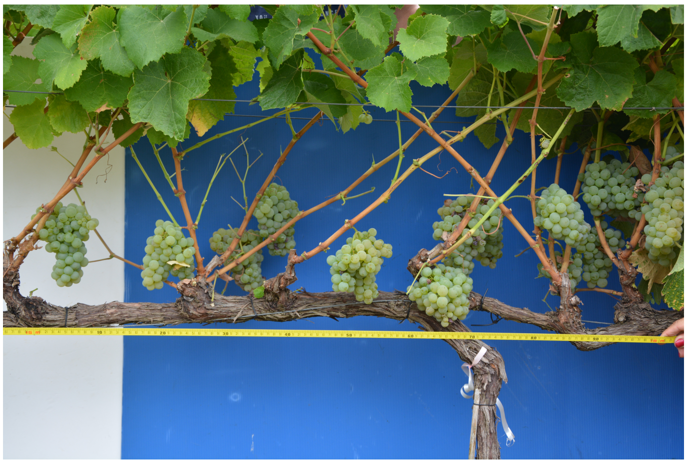


Figure 2.5: Example of defoliation in a vineyard patch

2.3 Summary

Works in yield estimation, yield components recognition, computer vision and machine learning in the agronomic context are discussed and presented. These projects were fundamental in the research for this work, but showed limitations, specifically the invasive techniques used, that this work aims to overcome. While aiming to not be invasive, other problems arise, such as the unknown number of occluded grape bunches. The way this problem is handled in this project is described in Chapter 3.5.3. Overall, this work provides a part of the project that brings a new, non invasive and automatic approach to the yield estimation problem.

3

Proposed System

Contents

In this chapter the developed system is explained in detail. In addition, some context is provided regarding the overall process. Firstly, an overview of the system, followed by an analysis of each step of the process: the pre processing of the data images; the segmentation process; the post processing applied to the resulting binary masks and a brief explanation of the final yield estimation model.

3.1 Materials

The mobile platform used is the Vinbot robot (Figure 3.1), that has in its mast a Red-Green-Blue (RGB) camera, at an adjusted height so it matches the canopy height, that is used to collect the image data. The robot is controlled by an operator with a controller and collects the data to an external hard drive directly connected to the robot.

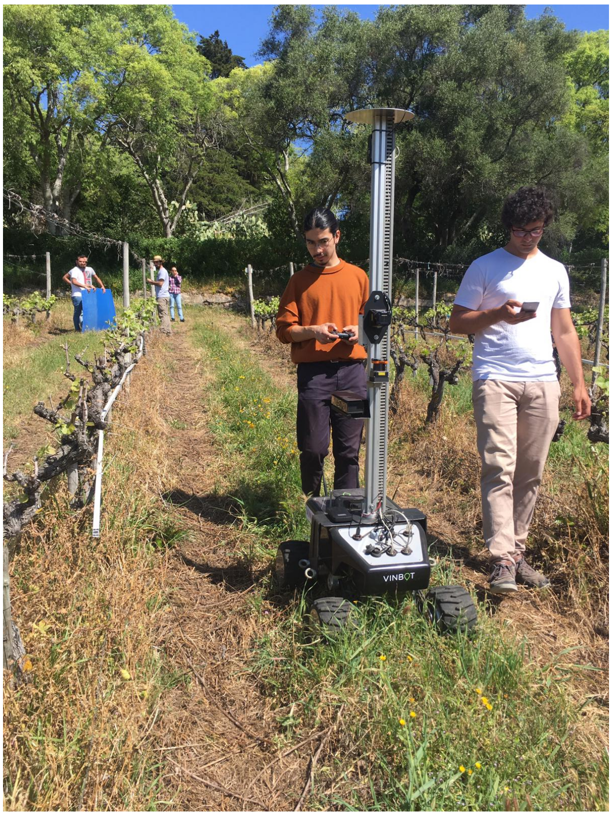


Figure 3.1: Vinbot robot driven by the Vinbot team, collecting data from the earlier stages of the vines development at the red variety vineyard of Instituto Superior de Agronomia (ISA)

The software used for the image labelling and other small tasks in the post processing is the ImageJ software. The programming was done in its entirety in python and the Neural Networks (NN) training was performed on the Google Colab online platform.

3.2 System Structure

The complete system is as follows(Figure 3.2):

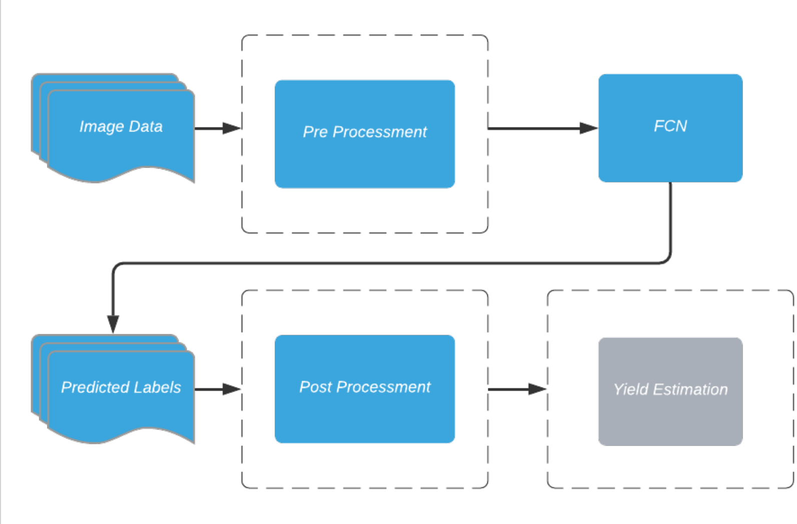


Figure 3.2: Complete system overview. In blue are the blocks that concern this work, in grey the system provided by ISA

Firstly, the robot passed through the vineyards with the RGB camera and takes approximately 1 meter wide images, meter by meter. These pictures are automatically pre processed, segmented and post processed. After the information is ready to be extracted from the images, this area of grape clusters in pixels is converted into cm2 and then into weight, estimating the yield.

3.3 Pre Processing

The objective with pre processing, in this case, is to transform the image data into a format that is more prone to learning, either by accentuating shapes or colours in images, for example so it would be easier for feature learning, or by simply formatting the image so it complies with the specificity of the segmentation algorithm.

The pre processing of the image data can be summarised in the following flowchart (Figure 3.3):

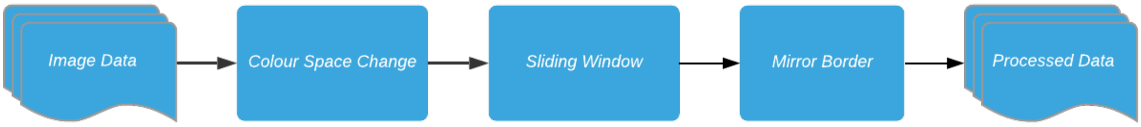
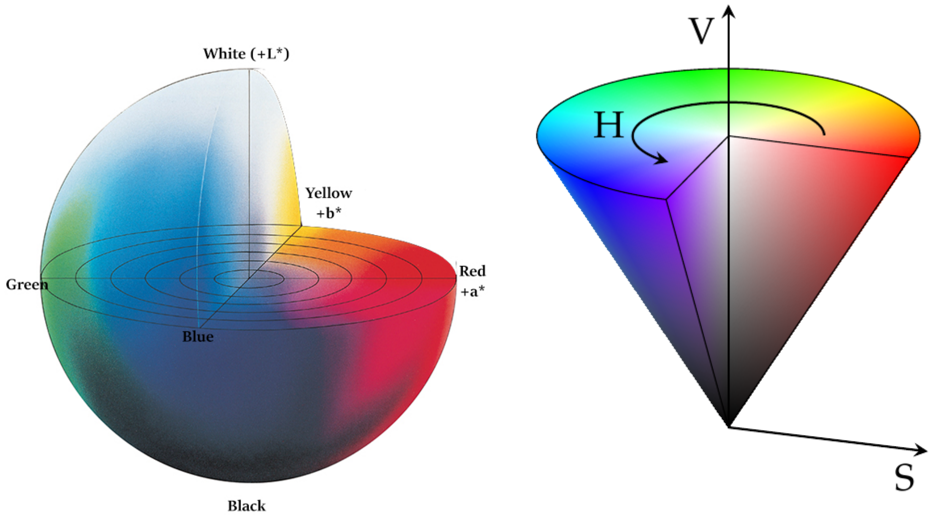


Figure 3.3: Pre Processing operations from the raw image to the processed network's input

3.3.1 Colour space

Although RGB is one of the most commonly used models in computer vision [5], it presents some disadvantages. The model produces a nonlinear and discontinues space, which makes the changes in colour hue difficult to pursue. This combined with the fact that the colour hue is also easily affected by illumination changes, makes that colour tracking and analysis a nontrivial task.

That being said, two colour spaces were considered to replace the RGB model, Hue Saturation Value (HSV) (Figure 3.4b) and Commission internationale de l'éclairage Lightness* a* b* (CIE LAB) (Figure 3.4a). Firstly, there is a model, RGB normalised, that dealt with one of the major problems, the sensitivity to illumination changes of RGB. The principle that guides this model is that a certain colour is formed using a certain proportion of three primary colours from the model and not a defined amount of each one. However, although it removes the aforementioned negative illumination effect, it also reduces object detection capability, due to the loss of contrast that the same illumination provides [5].



(a) CIE Lab colour space
source: [35]

(b) HSV colour space
Source: [36]

Figure 3.4: Colour spaces considered for the pre processing

The HSV model is independent to illumination changes, since it is enclosed in the value component (Equation 3.3) of the model. Solving this problem does not mean it does not have its own issues. One of them is when calculating the saturation component (Equation 3.2), a singularity caused when the $max(R, G, B)$(or value component) is zero, representing the colour black.

$$H = \begin{cases} 60 * \frac{G-B}{max(R,G,B)-min(R,G,B)}, R = max(R, G, B) \\ 60 * \frac{2+(B-R)}{max(R,G,B)-min(R,G,B)}, G = max(R, G, B) \\ 60 * \frac{4+(R-G)}{max(R,G,B)-min(R,G,B)}, B = max(R, G, B) \end{cases} \tag{3.1}$$

$$S = \frac{max(R,G,B) - min(R,G,B)}{max(R,G,B)} \tag{3.2}$$

$$V = max(R,G,B) \tag{3.3}$$

Another issue with this particular model is when the $max(R,G,B)$ and $min(R,G,B)$ values for RGB are the same, which corresponds to the grey tones. In this case the *hue* component is usually defaulted to zero (red colour). This may cause incorrect colour interpretations.

Perceptual colour spaces such as HSV also share a problem with the hue representation. Usually, it is represented as the angle of a circle, meaning that the restarting point of the circle (where the angle changes from 359° to 0°) causes a discontinuity in the colour hue, in this case the colour red. It is usually fixed by using two ranges for the hue at this position [5].

The CIE LAB model, on another hand, is based primarily on the physics aspect of light. CIE LAB is based of another CIE colour space model, the CIE XYZ. This model is calculated using the light wavelength from the physic representation of any specific colour. The equations that define this model are (equations 3.4-3.8):

$$L^* = 116 f\left(\frac{Y}{Y_n}\right) - 16 \tag{3.4}$$

$$a^* = 500\left(f\left(\frac{X}{X_n}\right) - \left(\frac{Y}{Y_n}\right)\right) \tag{3.5}$$

$$b^* = 200\left(f\left(\frac{Y}{Y_n}\right) - \left(\frac{Z}{Z_n}\right)\right) \tag{3.6}$$

where,

$$f(t) = \begin{cases} \sqrt[3]{t} & \text{if } t > \delta^3 \\ \frac{t}{3\delta^2} + \frac{4}{29} & \text{otherwise} \end{cases} \tag{3.7}$$

$$\delta = \frac{6}{29} \tag{3.8}$$

and Xn = 95.047, Yn = 100.0, and Zn = 108.883. These values are the tristimulus reference value for white for the CIE XYZ model.

The L^* component for this model encapsulates the illumination effect on the colours, providing a way to only remove the unwanted consequences of lighting changes. Other than this, the CIE LAB model basically generates a space where any colour is a combination of the a^* and b^* components, to a certain illumination intensity. The CIE LAB model can represent colours that are not handled by other models and, theoretically, it could represent an infinite number of chromatic combinations [5]. A more illustrative example is the RGB colour model representation in the CIE LAB space (Figure 3.5):

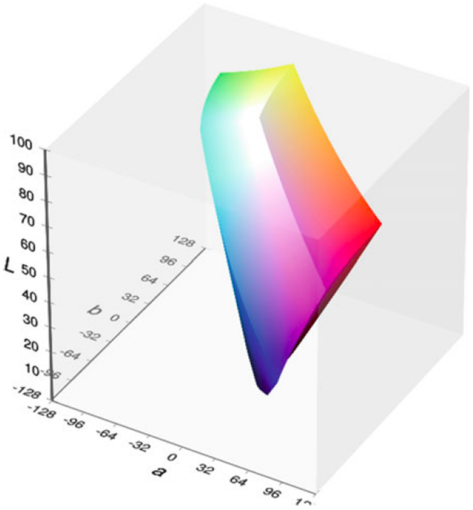


Figure 3.5: RGB colour space represented in CIE Lab colour space
source: [5]

With the arguments presented before, the colour space chosen for this project was the CIE LAB. The images are collected with a RGB camera and then converted into the CIE LAB colour space before any other action.

3.3.2 Image formatting

The next step in the system is the actual segmentation of grape clusters in the images. The algorithm selected has as input 572x572 pixel images, but produces an output of 388x388 pixel mask. Due to the nature of the algorithm, the frame pixels(from 388 to 572) will not be classified, since there is a resulting loss in border pixels from the convolutions in the NN . In order to not lose any data, the following steps were taken:

Firstly, a 388x388 pixel sliding window was passed through the image with minimum overlap (Figure 3.6).

Resulting in an image as Figure 3.7a, that will have to be enlarged to correctly correspond to the input size of 572x572 pixel. In order to do that a 92 pixel frame was applied, that mirrors the image.

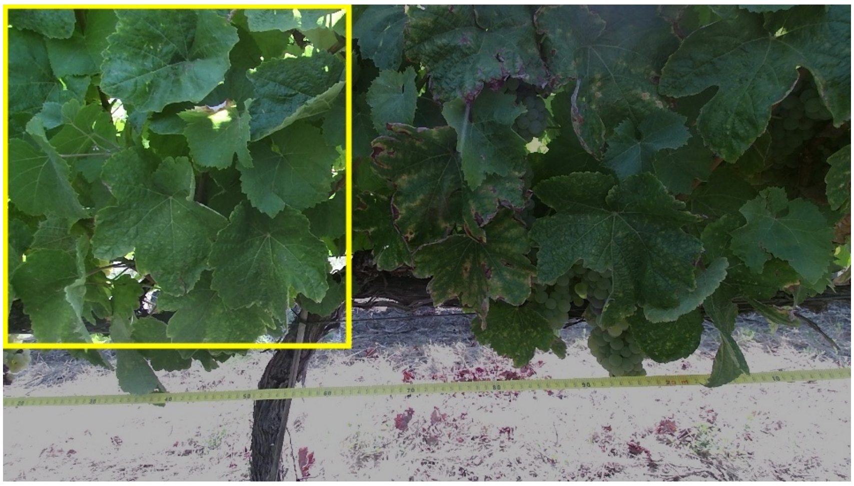
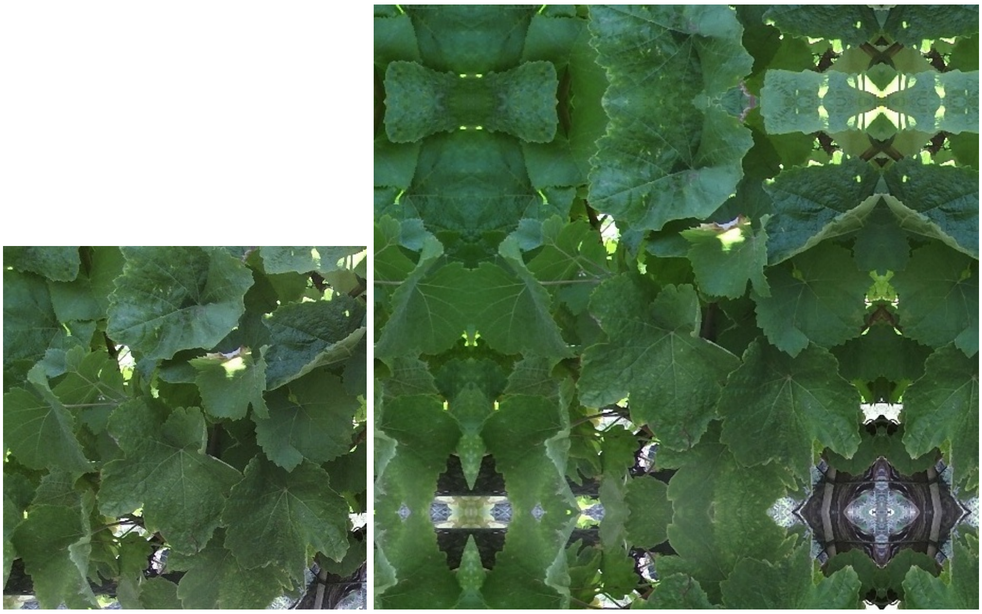


Figure 3.6: Example of an image with the first frame of the sliding window



(a) Resulting sliding window image

(b) Final image after mirror framing

Figure 3.7: Mirror frame transformation result

3.4 Segmentation

The image segmentation step of this system consists of a Fully Convolutional Network (FCN) that will take as input the previously processed images and will have as an output a binary mask of what is, or not, a grape cluster in the provided image. This specific format of NN was chosen due to the fact that its output, in this context, is the area of visible clusters for any specific image. By segmenting the image into bunch and background an area can be calculated by counting the number of pixels classified as bunch. This is a necessity for the final yield estimation model, that takes as one of its inputs an area of visible grape bunches. The architecture chosen for the task was the one used in [2], that will be studied in further detail in this section, along with a brief introduction to FCN. Another issue besides the one presented is that the classes that the networks aims to classify are imbalanced in the data set. In order to correct this imbalance the loss function was adapted. These changes are documented in the end of this section.

3.4.1 Fully convolutional networks

In general, a FCN can be comprised of two types of layers:

- Convolutional layers;
- Pooling layers.

The convolutional layer is the filter that, when passed through the input, defines what are the feature locations in a feature map. This is the main task of the network [37].

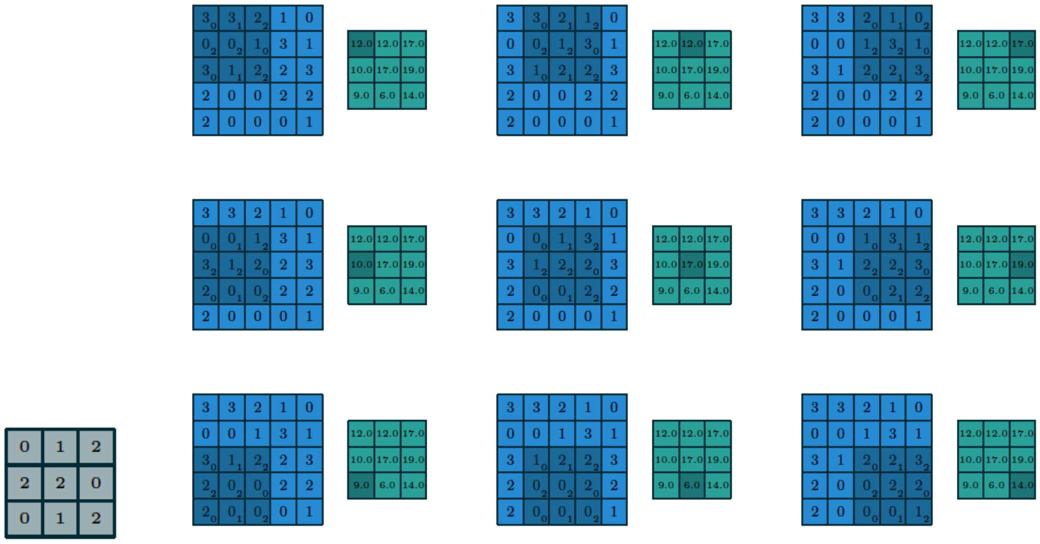
As stated in [6], this type of layer is composed of, essentially, a kernel that slides across the input feature map with a certain stride (distance between two consecutive positions of the kernel). In each position, the dot product is calculated. The resulting products are concatenated producing a new feature map as an output, as seen in Figure 3.8.

Another parameter that can be chosen, is the padding. Padding is the border pixels dimension that can be applied in the convolutional layer that can contribute to altering the size of the output. The output size is determined by Equation 3.9:

$$o = \frac{i + 2p - k}{s} + 1 \tag{3.9}$$

Where o is the output size, i the input size, p the padding and k the kernel size.

As for pooling layers, as said in [6,38], the objective in using them is to reduce space dimension and, consequently, computational power needed to process, to provide invariance to small translations of the input and to minimise overfitting.



(a) Kernel example

(b) Step by step convolution example

Figure 3.8: Example of a convolutional layer for feature extraction, where the grey square is the kernel, with 3x3 dimension, over a 5x5 input feature map with stride = 1, resulting in an 3x3 output feature map, represented by the green square.
Source: [6]

The principal behind the pooling layer is from a set of numbers create one that can be representative, according to the desired outcome. For example, one of the most used types of pooling is *max pooling*. Much like a kernel slides on an input map, this operation also has pre defined pooling window size and stride. Figure 3.9 exemplifies a *max pooling* operation with pooling window size of 3 and stride of 1 over a 5x5 input.

As described in [1], the structure of these networks are divided into two parts, the downsampling and upsampling. Both use convolutional layers, although with different purposes. The downsampling part is the only to use pooling layers. Firstly, the downsampling part extracts features from the input, reducing the input size at each layer. The upsampling part is where the final feature map that was calculated, combined with spatial data from the downsampling, reconstructs the input with the new learned information. This reconstruction is made possible through the use of transposed convolutions or up convolutions, which has the advantage of being able to carry out trainable upsampling. It provides as an output a reconstructed input of spatial dimension equal to the input of the correspondent layer in the downsampling part.

The combination of these types of layers is the basis of a FCN. Considering this type of network for this system, the U-Net [2] was chosen due to its positive results, further explained in 3.4.2.

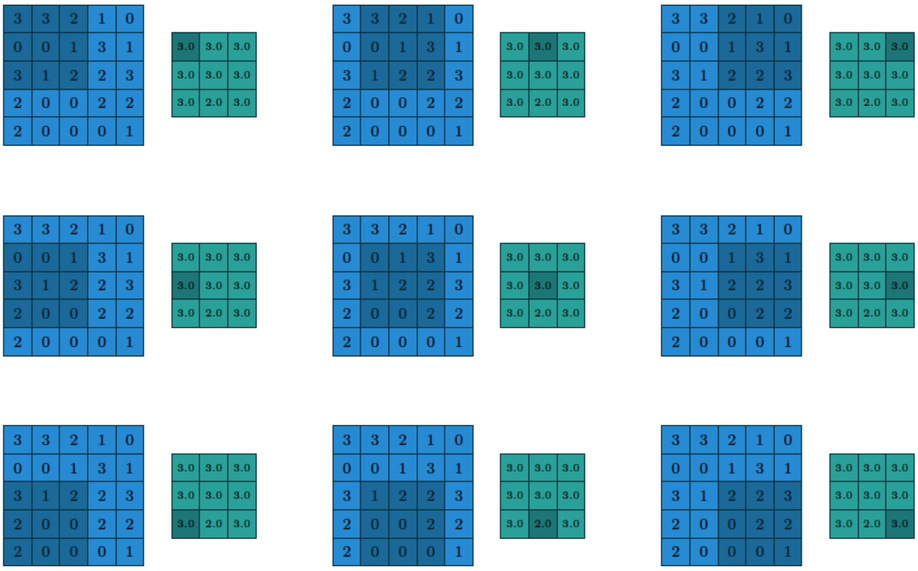


Figure 3.9: Example of a 3x3 pooling window over a 5x5 input feature map with stride = 1, resulting in an 3x3 output feature map, represented by the green square.
Source: [6]

3.4.2 U-Net

Considering the existing FCN, the U-Net stood out, since it showed positive results obtained with a similar deficiency of training data, which was solved with data augmentation. The U-Net network, as seen in Figure 2.2, was used in [2] for biomedical image segmentation.

This network is composed by two phases, a contracting and an expansive side. The contracting side is where the down sampling occurs and, correspondingly, the left side is where up sampling takes place. This network has as a base for every layer two 3x3 unpadded convolutions, each followed by a Rectified Linear Unit (ReLU) and a 2x2 max pooling operation with stride 2 for down sampling. At each down sampling step, the number of feature channels doubles. As for the right side, every step in it consists of an up sampling of the feature map followed by a 2x2 convolution, in which the feature channels are halved, a concatenation with the correspondingly cropped feature map from the down sampling path and two 3x3 convolutions, each followed by a ReLU operation. The cropping is necessary due to the loss of border pixels in every convolution. At the final layer a 1x1 convolution is used to map each 64 component feature vector to the desired number of classes. In total the network has 23 convolutional layers.

Given the limited amount of data that could be used for training, just 174 540x960 pixel images, validating and testing of any Machine Learning (ML) approach, the U-Net is a fitting choice for this work, since it provides the opportunity to be trained end-to-end with a small data set. This happens due to the reduced number of parameters to be trained, just above 30 million, less than state of the art networks such as Imagenet [33](60 million) and VGG [34](over 500 million parameters).

3.4.3 Loss function

In order for the network to consider the imbalance between classes it needs to have a weight factor in it. Using binary cross entropy defined in Equation 3.10,

$$H_b(p) = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot log(p(y_i)) + (1 - y_i) \cdot log(1 - p(y_i)) \tag{3.10}$$

just as is, was proven not to be enough, as shown in Chapter 4. As such, it was transformed into a weighted binary cross entropy (Equation 3.11):

$$H_b(p) = -\frac{1}{N}\sum_{i=1}^{N} w_c(y_i \cdot log(p(y_i))) + w_b((1 - y_i) \cdot log(1 - p(y_i))) \tag{3.11}$$

Where y_i is the label(1 or 0) and $p(y_i)$ is the predicted probability of that label. The weights w_c and w_b represent the weights relative to the class cluster and class background, respectively. It is the objective for the loss function to favour ”1” classifications. Therefore, the weight related to the cluster class must be higher than the weight related to the background class. According to [39], cost-sensitive learning regularly outperforms the other methods that deal with class imbalance. As such, this was the path chosen to deal with this problem. As for the weights, the values chosen must reflect the nature of the imbalance. Logically the weights should be an inversion of the relevance of each class in the image. As stated before, each image is 3.5% grape clusters and the remaining 96.5% is background, then the weights w_c and w_b will be chosen inversely, with $w_c = 96.5$ and $w_b = 3.5$.

3.4.4 Metrics

The aim of this project is to evaluate how well the yield estimation from vine images is made. As such, there will be two types of metrics. In a first stage is important to assess how well the segmentation is made in comparison to the ground truth and then, in a later stage, the areas of bunches predicted, after being put through the models that convert them into grape weight, should be rated against the correspondent ground truth.

Regarding the first stage, this implies, as stated before, a pixel-wise classification. Therefore, the metric that evaluates the success of the prediction must be one that is not binary but with an associated percentage. Moreover, each pixel has two possible classifications, grape cluster(1) or background (0). The metric should value more the correct classification of the first class rather than the second. This is important since the data set is imbalanced, having an average of 3.45% of an image occupied by grape clusters and a large percentage of images present a cluster classified pixel to total pixel ratio bellow or just over the average, as shown in Figure 3.10, where it is also represented that the majority of images(95%) have a 7.5% or lower ratio.

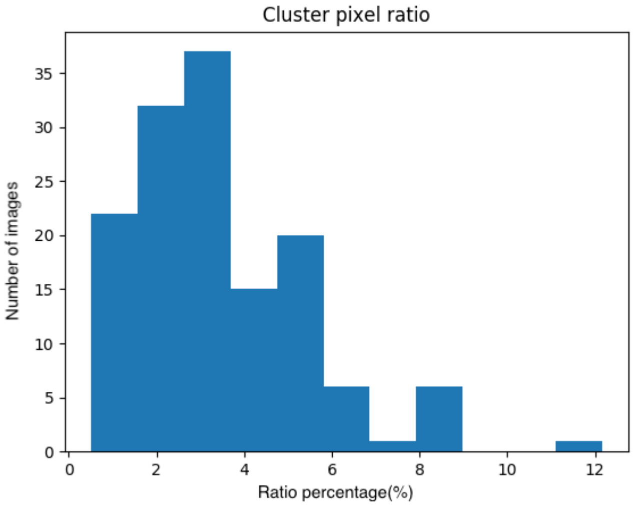


Figure 3.10: Cluster classified pixel to total pixel ratio. The y axis represents the number of images and the x the ratio bins in percentage

A metric that does not make the distinction could result in misleading evaluations. For example, if the metric considers only the overall correct or incorrect classification, as accuracy does, a prediction that classifies all pixels as 0 in an image with only 4% of pixels classified as 1, will result in an accuracy of 96%, which would not be reliable. That being said, the metric used was the Intersection Over Union (IOU), also known as the Jaccard index(3.12). This metric has the advantage of depending only on the classification of the chosen cluster class. It provides the ratio of correctly classified pixels by the total of positives in both predicted and ground truth, as exemplified in Figure 3.11.

Moreover, this metric is also used in other segmentation evaluations, such as object detection [40] and dermoscopic image segmentation [41].

$$IOU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \tag{3.12}$$

The second stage of the evaluation is further explained in Chapter 3.5.3.

3.5 Post Processing

Given the presence of some noise in the image, i.e. small false positives clusters and the natural order of the system, post processing operations were made. They removed very small clusters of false positives

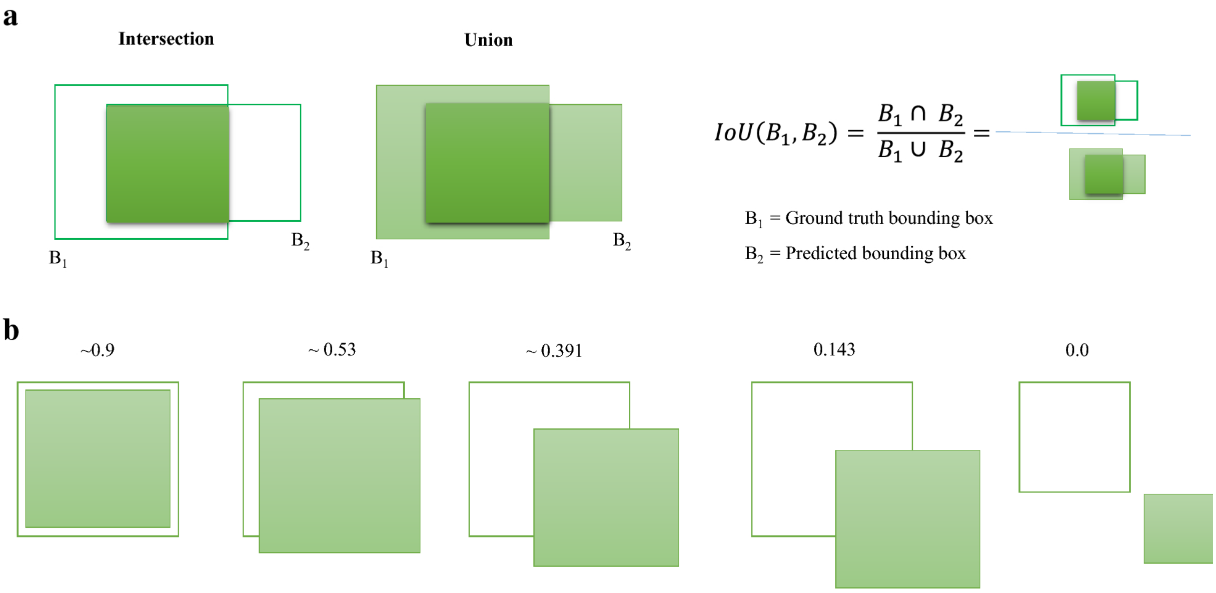

a
Intersection
Union
B1
B2
B1
B2
$IoU(B_1, B_2) = \frac{B_1 \cap B_2}{B_1 \cup B_2} =$
B1 = Ground truth bounding box
B2 = Predicted bounding box
b
~0.9
~ 0.53
~ 0.391
0.143
0.0

Figure 3.11: (a) provides a visual explanation of IOU and (b) shows scores examples for the metric. Source: [7]

through the use of morphological operations, further explained in 3.5.1. Following the operations, the image was reconstructed. In this step, false positives are also eliminated due to the overlapping areas that resulted from the sliding window, which allow for two or more predictions for some part of the image. With the reconstructed image, with the help of a scale in the image, the area of visible bunches in pixels was converted to cm^2, which is later converted into a prediction of the total area of bunches, including the non-visible ones, using a canopy porosity based model developed in [17] and further explained in Chapter 3.5.3. Finally, this total area can be converted into weight, resulting in the yield estimation for the area in study.

This being said, the post processing can be divided into three smaller steps (Figure 3.12).

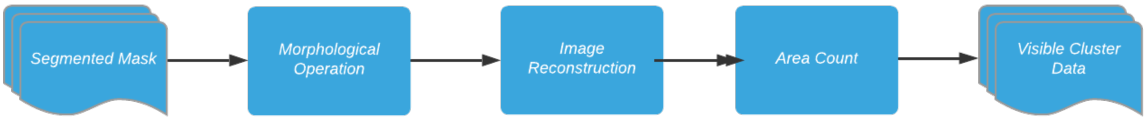


Figure 3.12: Post processing system architecture

3.5.1 Morphological operations

Morphological operations are among the most common binary image operations [8]. One of the uses of these non-linear filters is removing small objects in an image and smoothing borders.

These operations consist of, firstly, convolving the binary image with a structuring element. These structuring elements can be of any shape, from a 3x3 matrix to any more complex structures.

The standard morphological operations are (Figure 3.13):

- dilation: dilate(f; s) = θ(c; 1);
- erosion: erode(f; s) = θ(c; S);
- majority: maj(f; s) = θ(c; S=2);
- opening: open(f; s) = dilate(erode(f; s); s);
- closing: close(f; s) = erode(dilate(f; s); s).

where,

$$\theta(f,t) = \begin{cases} 1 & \text{if } f > t \\ 0 & \text{otherwise} \end{cases} \tag{3.13}$$

and S is the stride of the structuring element, s is the structuring element, f the image and c the integer-valued count of the number of 1s inside each structuring element as it is scanned over the image, resulting of Equation 3.14.

$$c = f \otimes s \tag{3.14}$$

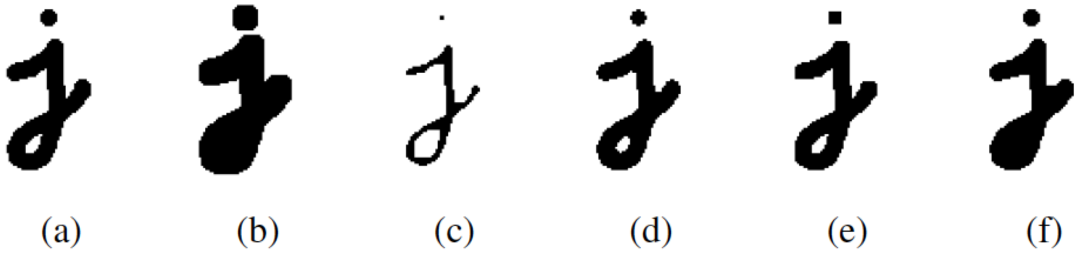
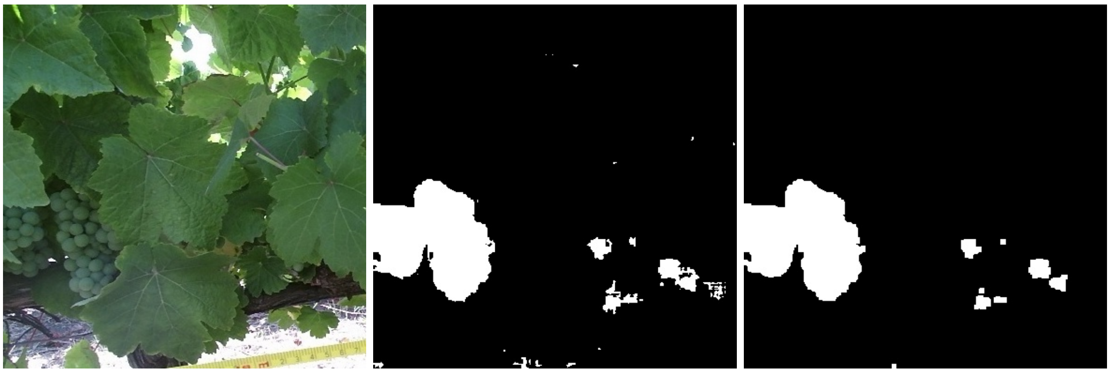


Figure 3.13: Examples of morphological operations: (a) original image; (b) dilation; (c) erosion; (d) majority; (e) opening; (f) closing.
source: [8]

As it is possible to observe from Figure 3.13, dilation thickens the object, while erosion thins it. The last two operations, open and close,tend to leave large regions and smooth boundaries unaffected, while removing small objects or holes and smoothing boundaries. It was these characteristics that made the ”open” operation the chosen for the first step of the post processing. An example of the effect the operation has on the processed image is shown in Figure 3.14.



(a) Original image **(b)** Resulting mask **(c)** After open operation

Figure 3.14: Morphological operation applied on a processed image

3.5.2 Image reconstruction

As stated before, the existence of overlapping areas in the predicted images allows for another mean of dealing with false positives. The way to take advantage of this situation is by creating a decision system for the different areas of overlapping in the reconstructed images regarding what is or not a grape bunch. The image presented in Figure 3.15 illustrates the overlapping segments, where the number in each segment indicates how many images overlap:

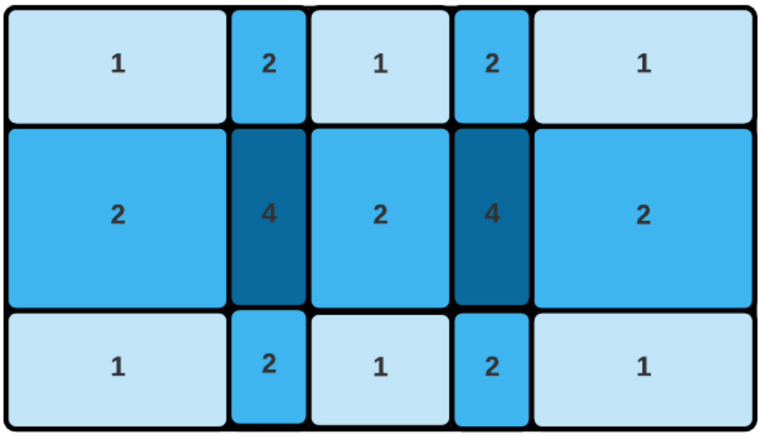


Figure 3.15: Regions of overlap in the reconstructed image:"1" is where there is only one sliding window image that classifies that segment; "2" represents that the segment in question is classified by two images;"4" is where the segment is classified by 4 different images

For the "1" regions, the prediction of that image stand for the final reconstruction. For the "2" regions, both predictions must agree on a pixel level, so a *logical and* operation is applied. As for the "4" regions, a 75% certainty is believed to be enough to classify a certain pixel as part of a bunch. Although the problem is an excess of false positives, false negatives also occur. The three out of four approach provides some robustness against false negatives in the reconstruction in opposition to a strategy that is simply based on a *logical and* approach.

3.5.3 From counting pixels to yield estimation

After the reconstruction, the following step in the process is to account for the number of pixels classified as grape bunches in each image in order to start the process that takes the area of clusters identified and converts it into kg. This process starts with calculating the ratio from pixel to cm^2. Since the Vinbot project has not been fully automated, this task is performed by hand, image by image. With the ratio, it is just a matter of transforming the area in pixel into an area in cm^2. This area of visible bunches will then be used as input into the previously developed models.

These models, developed in [17] and also used in [9, 42], take part in two stages of this process (Figure 3.16):

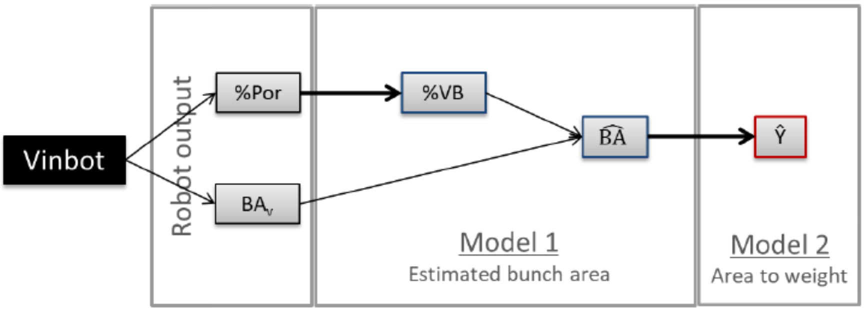
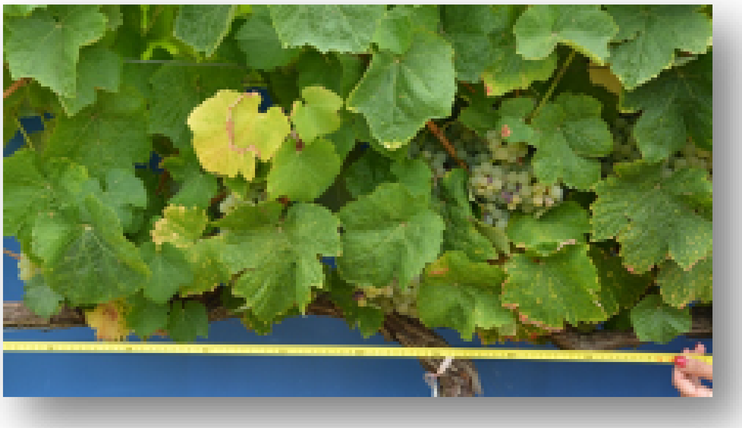
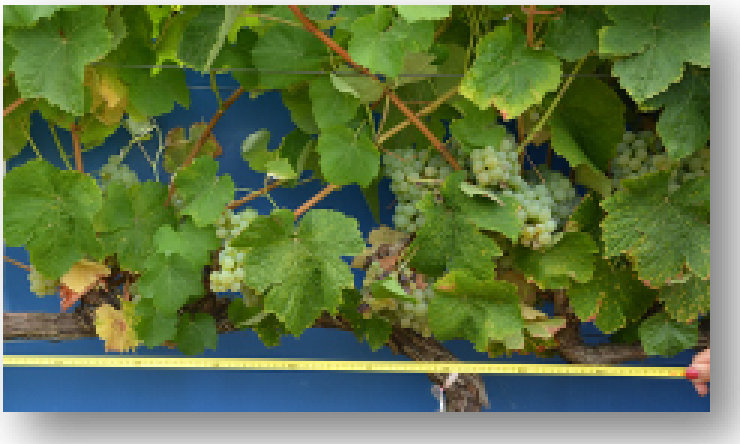


Figure 3.16: Yield estimation diagram from the Vinbot output until the final estimation of yield per image. %Por=bunch zone canopy porosity (in percentage); BA_V = visible bunch projected area; %VB = percentage of visible bunches; $\widehat{BA}$ = estimated bunch projected area (total per image); $\hat{Y}$= final yield estimation per image.

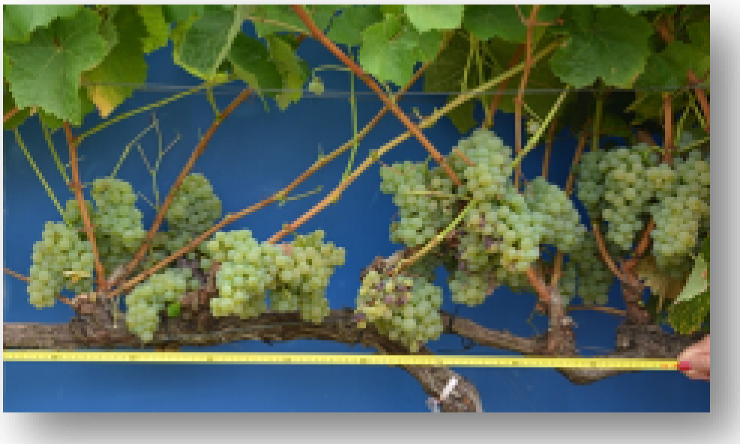
The first model is used to estimate the visible bunch percentage of the total existing bunches, taking into consideration the porosity of the vine. The porosity is correlated with the percentage of visible bunches, since the major cause for bunch occlusion is leaves from the vine. Therefore, the higher the porosity percentage (more empty spaces in the canopy) the lesser the leaf occlusion, as it is seen in Figures 3.17a, 3.17b and 3.17c. Knowing the area of visible bunches and the percentage to which it corresponds, due to the porosity correlation, it is possible to make a projection to the total bunch area, visible and covered. This projection is then fed into the second model to calculate the final weight, transforming cm^2 to kg through a polynomial fit where the bunch area is the independent variable and the weight the dependent one. Although it may seem that this transformation induces error, it has been shown in [43, 44] that grape bunch area is a good predictor of bunch weight, with different parameters depending on the variety and maturation stage of the grape. These parameters vary since the bunch compactness also changes from variety to variety [45], resulting in different weights for the same projected area and the berry size is also different depending on the maturation stage.



(a) Untouched vine with **16% porosity** and **67% occluded bunches**



(b) Defoliated vine with **25% porosity** and **50% occluded bunches**



(c) Defoliated vine with **48% porosity** and **0% occluded bunches**

Figure 3.17: Example of progressive defoliation of a vine canopy

3.5.4 Evaluation

The second stage of the evaluation process is performed after these models are applied. To assess the estimation's precision, the percentage of the relative error(Equation 3.15) to the ground truth was calculated .

$$RE(\%) = \frac{|Actual\ yield - Estimated\ yield|}{Actual\ yield} \times 100\% \tag{3.15}$$

This second evaluation is important since there are steps in between the output of the segmentation network and the final estimation, as previously explained. This metric provides an overview of how good is the data set (if the data collection process needs to be adapted in future takes and if the not automated parts of this system are done correctly), evaluates the robustness of the models with different sourced data (automatic and hand made) and suggests a path for the project to take moving forward.

3.6 Summary

In this chapter the overall system was presented with different focus in each step of the process.

Firstly, the main system is shown and explained through Figure 3.2. Then, by order, the subsystems are elaborated in further detail, with the necessary theoretical background associated with each step. Starting with the pre processing, where there is a colour space conversion, from RGB to CIE LAB, followed by a sliding window to produce accurate sized images for the next step. After the image has been processed, it is fed into the segmentation phase, where a FCN, the U-Net, will classify at a pixel level what is and not a grape bunch in the image, producing a segmentation mask. This mask is then further tweaked in the post processing stage. Given the tendency to produce false positives, mostly small clusters, the morphological operation open is applied, removing part of the false positives. The image is then reconstructed from the pieces made by the sliding window. This reconstruction may also remove some of the remaining false positives through comparison between masks when overlapping exists. Finally, the image's pixels are counted and converted into cm^2, which will be combined with the previously calculated porosity of that specific image, projecting a total cluster area with the model designed for this purpose. This area is converted into weight using another model. After processing all images, the summed weights result in the final yield estimation.

4

Experimental Results

Contents

This chapter addresses the experiments and decision making process that led to the system shown in Chapter 3.

Firstly, a description of the existing data set is given, followed by the description of the metrics considered for the segmentation evaluation and overall experimental setup.

Secondly the experiments performed are explained, starting with the baseline. This is used as a basis of comparison to the remaining experimental setups and results.

4.1 Data Set

4.1.1 Instituto Superior de Agronomia (ISA) vineyard and characteristics

ISA, being agronomy their main field of study, has vineyards for the study of viticulture and oenology. This vineyards contain seven white varieties of grape, Macabeu, Moscatel Galego, Moscatel de Setúbal, Alvarinho, Viosinho, Encruzado and Arinto [46]. They also have four red varieties, Touriga Nacional, Trincadeira, Cabernet Sauvignon and Syrah [47]. From all these varieties, two were selected for further study in this work, Encruzado and Syrah. The total vineyard cultivation area is composed by one hectare for red varieties [47] and 1.7 hectares for the white varieties [46],the last one shown in Figure 4.1.

Figure 4.1: ISA white variety vineyard overview
Source: [9]

The available images for training, testing and validating were taken in 2018, with a total number of 174 540x960 pixel images, as the one shown in Figure 4.2, corresponding to the last two phases of the grape's maturation, veraison and harvest.

The images were passed through a sliding window of 388x388 pixels (shown in Figures 3.6 and 3.7), the output size for the U-Net, with overlapping for training and without for the validation and testing data set. To match the network's input size, a mirror frame was applied with 92 pixels to each border.

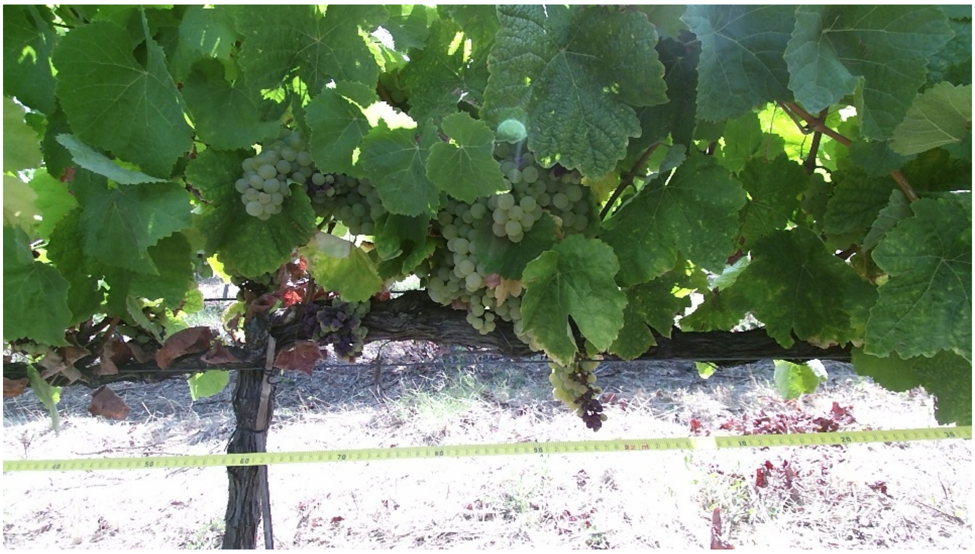


Figure 4.2: Example of vine segment image

This frame stops the loss of border information by the cropping performed by the network. Also, to increase the use of the data available, data augmentation was used. The only data augmentation technique used was the image mirroring in the vertical axis. Other image operations including, but not limited to, horizontal flipping, small deformations and rotations would not make sense since there are no representation of the resulting images in nature. An example of this is that clusters do not grow upside down, against gravity.

As for the data distribution, 80% was used for training, 10% for test and 10% for validation. After dividing the images between the different sets, data augmentation and image formatting operations were performed on the training set. The aforementioned operations were different for the training and the remaining sets, since no data augmentation could be applied in the test and validations sets, tainting the results. So, for the training set, a siding window, as mentioned before, was passed trough the image, creating 6 388x388 pixel images for each 540x960 pixel image. Then, that number doubled with the inversion of each image around the vertical axis. Therefore, each 540x960 pixel image created 12 388x388 pixel images. Besides the pre processing of the training set, the other two sets could only generate 2 388x388 images per original image, given that more than 2 would result in overlapping. With this information, the data can be summed in Table 4.1:

Table 4.1: Number of images of each data set before and after data augmentation and image formatting operations: training, test and validation

	Number of images(pixel area)	
Data set	Before data augmentation(540x960p)	After data augmentation(388x388p)
Training	140	1680
Test	17	34
Validation	17	34

Even though the test and validation sets present a lesser number of images, they are representative of the task at hand, since the images belonging to those sets were selected from regions of interest, as explained further in the next section.

For the second part of testing, the yield estimation evaluation, the 2019 data set was made available. This data set is comprised of 4 different testing conditions, 2 grape varieties, Encruzado and Arinto, at two stages of maturation, veraison and harvest, with a total of 40 meters per combination, resulting in 160 meters of vine over 197 540x960 pixel images. These images do not have a useful ground truth for segmentation purposes, therefore they are not used in that sense. This division between segmentation testing and yield estimation evaluation in the 2 sets is further explained in detail in Section4.3.

4.1.2 Test and validation sets

Although a sliding window passed through the training set, the same could not be for the test and validation data sets since the sliding window was not without overlapping in the images that it produced, as it is possible to see in Figure 4.3.

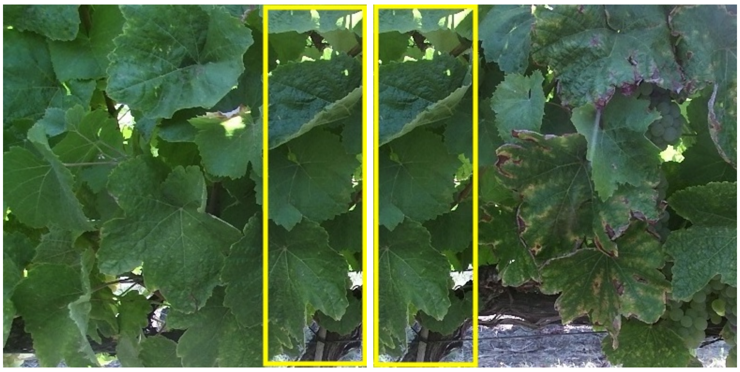


Figure 4.3: Example of the existing overlapping

If there was any overlapping in the test and validation sets, at some point the same area would be evaluated twice, producing unrealistic scores. Given the dimensions of the image, only two 388x388 pixel images could be extracted from a single 540x960 pixel image. To decide from where in the image they would be extracted, the set was analysed to see where were the parts where the largest incidence of grape clusters, from now on refereed to as regions of interest, so that the segmentation machine would be properly evaluated. To get to the incidence information the ground truth masks were overlapped, element wise summed and normalised, resulting in Figure 4.4, where the whiter the pixel, the more grape bunches over the set.

From this image, the data sets for testing and validating were chosen, always with the same positioning. The regions of interest were not taken into consideration when producing the training data set. Even

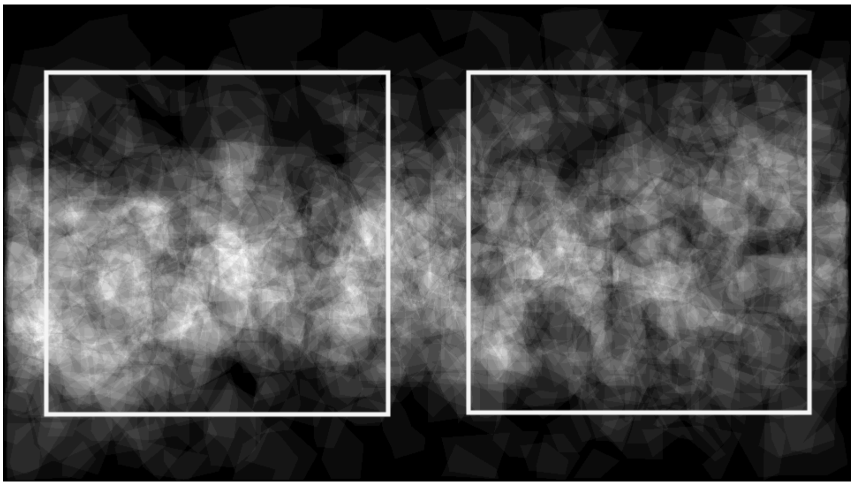


Figure 4.4: Grape bunch distribution over the test and validation data sets. The grey scale indicates the incidence of bunches in any given area. The white squares were the areas chosen for the sets

though the evaluation metrics are possibly penalised by this choice, the trade off was between taking the regions of interest into consideration or having a larger data set. Given the data shortage of the overall set, the regions of interest information could not be taken into consideration for the training set.

4.2 Experiments

All experiments that were made that contributed to the decision making process behind the final system are explained in this section. All experiments were performed using the Google Colab platform, with a GPU accelerator. Given that every time any experiment was made it could be done with any number of different available GPUs, there is no possibility for time comparisons between the experiments. Also, the number of epochs used for each experiment always had the limitation that each session on the platform could only last 12 hours maximum, disconnecting after that. This was a problem for the training, by the fact that it was limited to a certain extend.

4.2.1 Naive Baseline

As mentioned before in Chapter 2, the U-Net neural network has as perks to its use the fact that it can be trained end-to-end with smaller data sets such as the one available and present satisfying results [2]. This was one of the main reasons behind its choice as the segmentation network for this work. This baseline experiment used the U-Net as it is presented in Chapter 3, trained with binary cross entropy as the loss function, since that in [48] is stated that cross entropy may be more robust in maintaining its performance advantage for problems with limited data when compared to a squared-error function, no data augmentation, no pre or post processing of any kind and for 100 epochs.

This experiment resulted in all the images being 100% classified as background, with a mean of 96% accuracy and 0% Intersection Over Union (IOU). From this, several hypotheses were made regarding the failure of the experiment:

- The loss function is not taking into consideration the existing class imbalance in the images;
- There is not enough data for the machine to learn from;
- The images may need some previous processing before being used for training.

4.2.2 Loss function

As explained in Chapter 3, the loss function was tweaked. Although there is a rationale behind the combination proposed previously, other weight combinations were tested. The results for the validation set during the 100 epochs of training are shown in Figure 4.5:

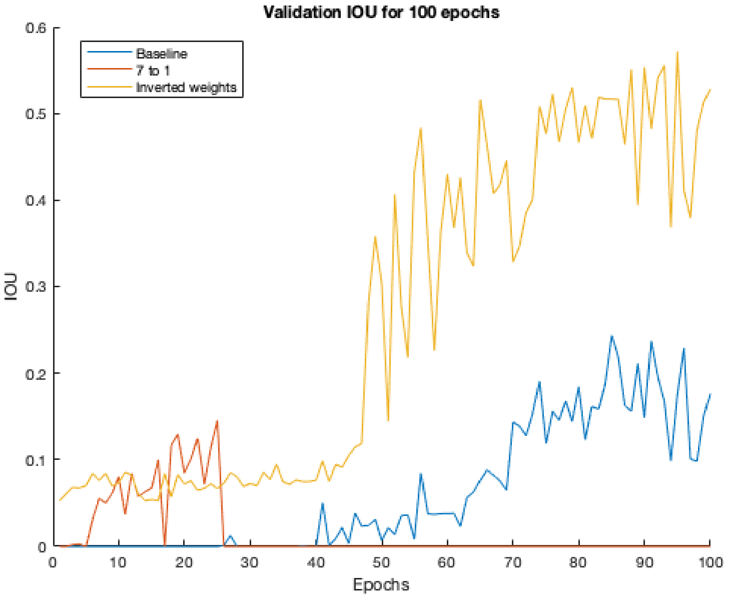
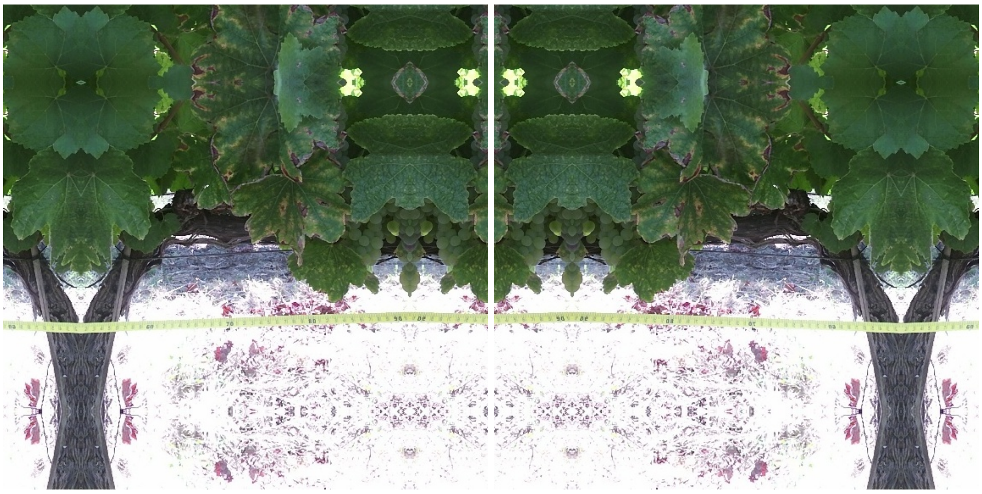


Figure 4.5: Evolution of the validation IOU over 100 epochs for three different sets of weights

With this experiment is possible to conclude that the best combination for the weights it is the inversion of the ratio of their imbalance, as it was expected in theory.

4.2.3 Data augmentation

As previously done in [2], to solve the problem of insufficient data, data augmentation was performed. The image was then inverted over the vertical axis, as it is possible to see in Figure 4.6a and 4.6b.



(a) Original image

(b) After mirroring

Figure 4.6: Example of the data augmentation performed

This data augmentation technique led to better values in the metrics used, especially during the first 50 to 60 epochs. Using the weighted binary cross entropy with the weight set defined in the last section(4.2.2), there was a clear improvement (Figure 4.7).

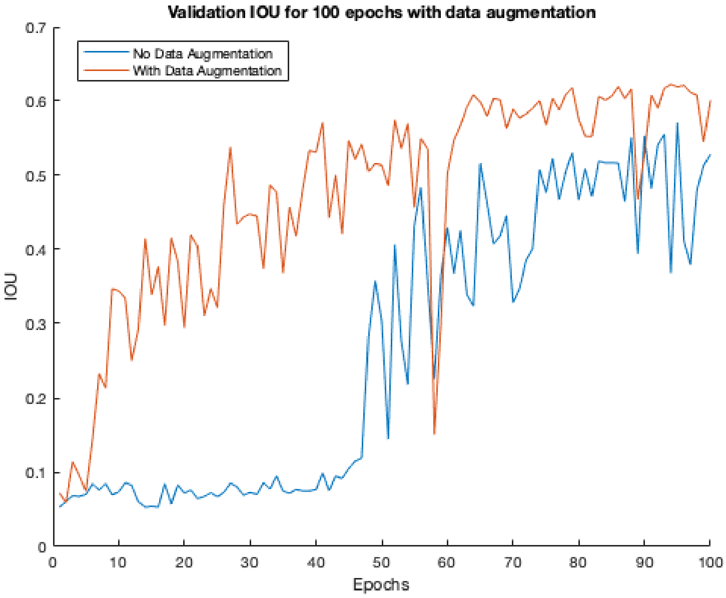


Figure 4.7: Evolution of the validation IOU over 100 epochs for the set with and without data augmentation

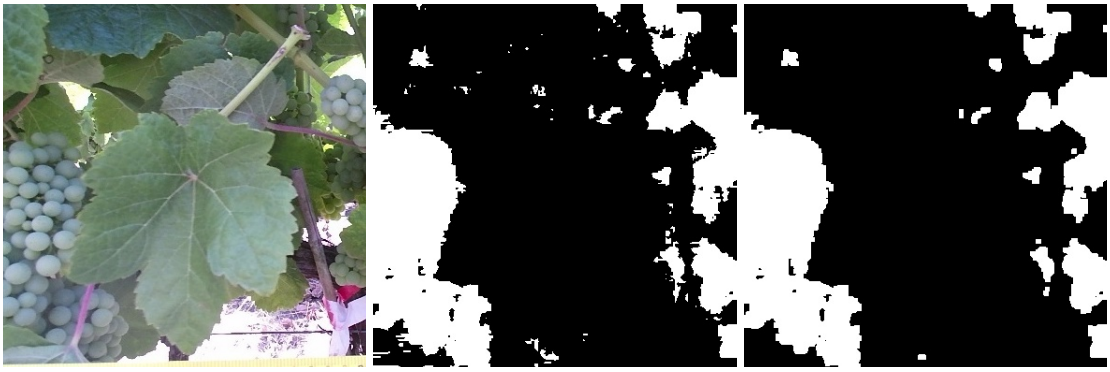
The use of data augmentation resulted in a score that was higher than the previously best, achieving 62% validation IOU over the 57% obtained without the augmentation. Following the augmentation operations, the next step of experimentation is focused on the stages before and after the segmentation. As for the pre operations, the colour space is changed. After the segmentation, the morphological operations are tested.

4.2.4 Colour space

As stated before in Chapter 3, the used colour space for this work was the Commission internationale de l'éclairage Lightness* a* b* (CIE LAB). Besides the advantages of the use of this colour space that were already described, testing was also performed for empirical confirmation. The results showed that when using this colour space, for the same test set, the results improved when compared to the Red-Green-Blue (RGB) colour space, going from 49% IOU to 62%. It is possible to assume that this improvement comes from discarding the negative effect of illumination, as explained before in Chapter 3.

4.2.5 Morphological filters

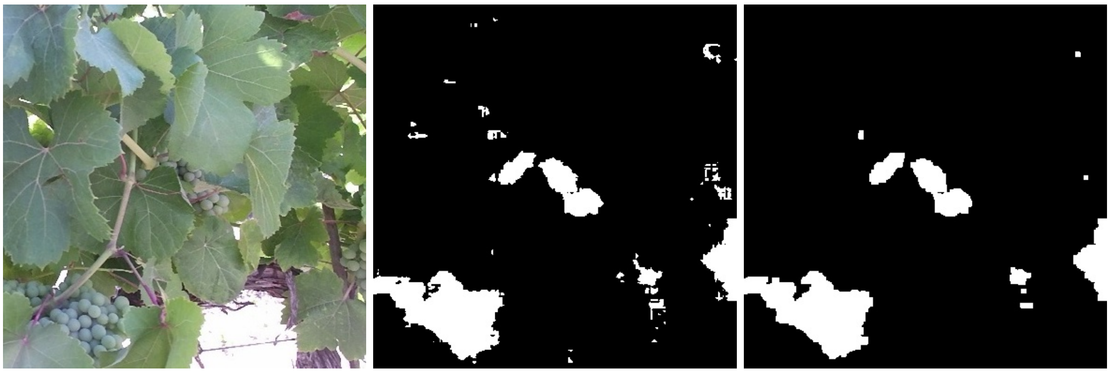
The use of morphological filters, as it is mentioned before in Chapter 3, is mainly to reduce the number of false positives in the image, mostly related to small miss identified clusters. The ”open” operation was able to remove a significant part of these clusters and smoothed the grape bunch boundaries, also increasing the IOU. In the first example, Figure 4.8, the IOU increases from 70% to 72% and in the second example, Figure 4.9 it presents an IOU of 58% that transforms into 61% after the morphological operation. Although the amount of pixels removed may be small, the difference in the metric is notorious, demonstrating the sensible nature of the problem.



(a) Original image **(b)** Resulting mask **(c)** After open operation

Figure 4.8: First example of the morphological operation applied on a processed image

After all the testing, the overall process was changed according to the results. The test set produced the following scores for the described experiments (Table 4.2):



(a) Original image **(b)** Resulting mask **(c)** After open operation

Figure 4.9: Second example of the morphological operation applied on a processed image

Table 4.2: IOU scores for the grape bunch class in the test set in every experiment

Experiment	Test set IOU results
Naive Baseline	0%
Loss function	47%
Data augmentation	49%
Colour space change	62%
Morphological filters	64%

4.3 Yield estimation using the 2019 data set

With the testing made for the first stage of evaluating the system, answering the question of how well does the network's segmentation match the ground truth, the next step is evaluating the whole system for its purpose, yield estimation. Although the segmentation evaluation through the IOU metric is a good indicator of how well the system is behaving, the most important metric is the final yield estimation after the models are applied. For this purpose another data set is available for evaluation, the 2019 set. Until this moment only one data set was used, the 2018 set, used for training and evaluating the segmentation, but the new 2019 set can be used for the second stage of evaluating.

In 2018, the project group used a different model than the one used in 2019. This model did not take into consideration the vine's porosity for the estimation and as a consequence no porosity data is available. Without this data, it is not possible to use this 2018 set for the final yield estimation evaluation. As for the 2019 data set, it was not possible to use it for the segmentation evaluation since it does not have a complete segmentation ground truth, only the final weights. In this context, the ground truth is only partially available. Since the estimation is made meter by meter and a frame contains more than a meter, the images were manually cropped. The segmentation was made only for the cropped images. It would be extremely difficult, unlikely and time consuming to crop the networks segmentation to the match the ground truth. Attempts were made but all failed, since if not exactly match for all images, the results would not be trustworthy. Other than this, for some of the images the ground truth is missing from the set. This is the reason why both sets could not be used for both evaluations.

For this experiment, four situations were examined, two varieties, Encruzado and Arinto and two stages, veraison and harvest, the last two stages of the maturation process. Besides the test of how well it estimates yield, this experiment also tests the robustness to other varieties, since it was trained only with Encruzado images and if it can provide a decent estimation (less than 10% error, stated in [49] to be the norm) from a few days before the harvest to three months ahead in the veraison stage. It also allows for testing and better adapting the not automated processes to the automation of the segmentation step. All data sets, for both Arinto and Encruzado, are composed of several images of four Smart Points (SP), each a set of ten consecutive meters. The SP are the same between maturation stages. The images used are not completely untouched, since the data for the porosity percentage had to be collected with a blue background, given it is not yet an automated process, and there is a scale present in all images to divide the meters of the SP and to provide the scale used to convert pixels into cm^2.

It was expected that the harvest results would be more precise than the veraison, since the data was taken closer to the actual harvest, inducing less error in the models. This was proven not to be true with the veraison sets showing promising results, within the 10% objective, and the harvest sets resulting in a complete miss estimation. The possible reasons behind the results are explained in the end of this chapter.

4.3.1 Veraison

As for the veraison analysis, the results are very satisfactory. Observing Table 4.3, the Encruzado variety has only 3% relative error, just over 2 kg over the actual yield and the Arinto variety has 10% relative error, 13 kg less than the ground truth. Although there are some discrepancies in the meter wise analysis presented in Figures 4.10 and 4.11 (where,for example, meter 12 is meter 2 of SP2), especially where the actual yield is unexpectedly low or high, as is in meter 9, 11 and 31(low) and 1,10 and 38(high) for the Encruzado results and meters 2, 11 and 13(low) and 28, 37 and 40(high) for the Arinto results, the final result shows that the underestimation compensates the overestimation. Also, as it is possible to observe from Figures 4.12 and 4.13, the illumination conditions in both are constant through out the canopy and there are not shadows on the ground, which could contribute to incorrect segmentation and, consequently, incorrect estimations.

Table 4.3: Final results for the yield estimation in kg in veraison stage with the associated relative error to the actual yield between parenthesis

	Encruzado	Arinto
Actual yield	69,2 kg	123,4 kg
Hand segmentation	67,6 kg(2%)	121,3 kg(2%)
Automatic segmentation	71,0 kg(3%)	110,6 kg(10%)

The meters mentioned before show the most relevant errors in estimation in all the studied area. After analysing the data for each meter is possible to conclude that the discrepancies were caused by the model and not by the segmentation, since the hand segmentation produced similar estimations to the automatic. Meter 9 had the same estimation of 1.2kg for both segmentatios and in meter 11 the estimations varied 0.1kg, for example. Meter 1 of the Encruzado set had for the manual segmentation 2.5 kg and for the automatic 2 kg and meter 38 had a difference of just 0.08 kg.

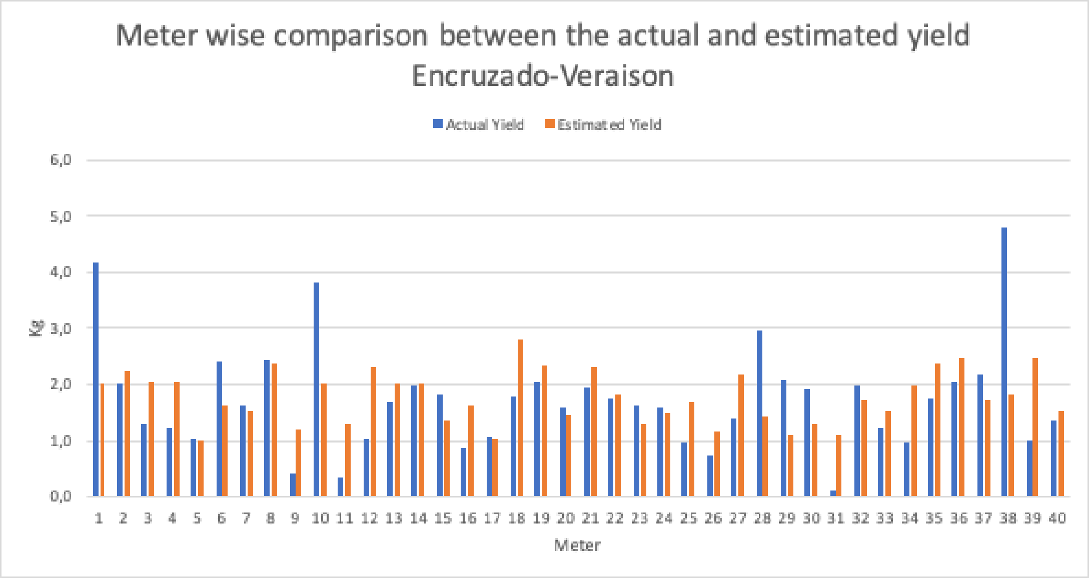


Figure 4.10: Meter by meter analysis of the results for the Encruzado variety in the veraison stage for the automatic segmentation

As for the Arinto results, the overestimation in the previously mentioned meters was caused by miss classifications in the network, which is prone to produce false positives, as said before. In the underestimations, again the meters mentioned had similar estimations to the hand segmentation ones, all of them with less than 1 kg of a difference. These last errors in estimation are presumably caused by the model.

The segmentation network seems robust enough so it can extend to the Arinto variety, even though it was only trained with Encruzado images. They are both white varieties, but with some differences. This limited similarity is what may explain the error passing from 3% to 10%, resulting in a good estimation for the Arinto but not as good as the Encruzado's. This is possible to deduce from comparing the quality of the segmentation masks in Figures 4.12 and 4.13.

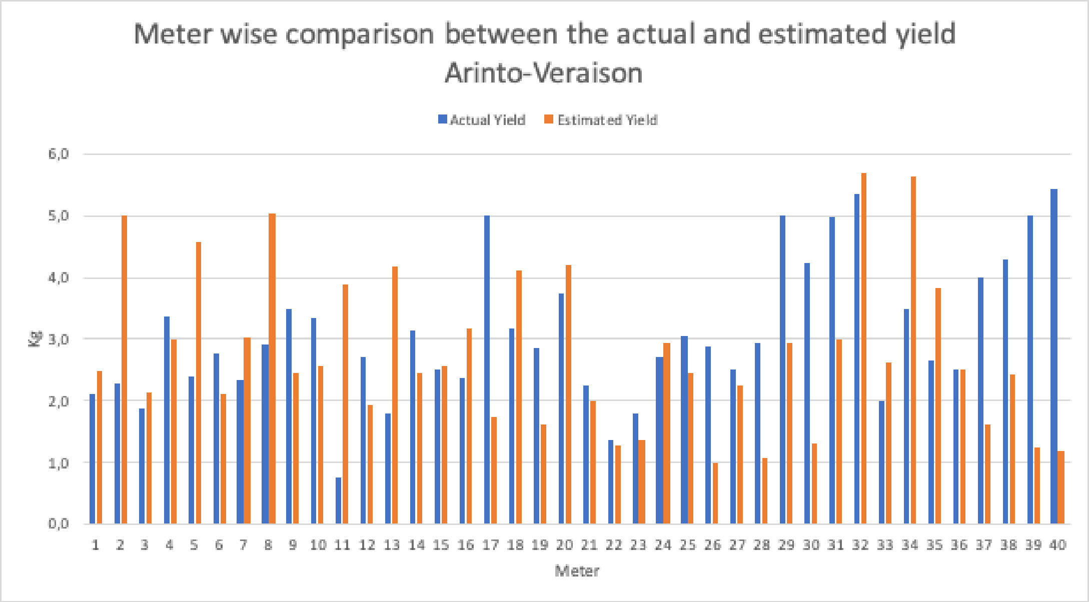
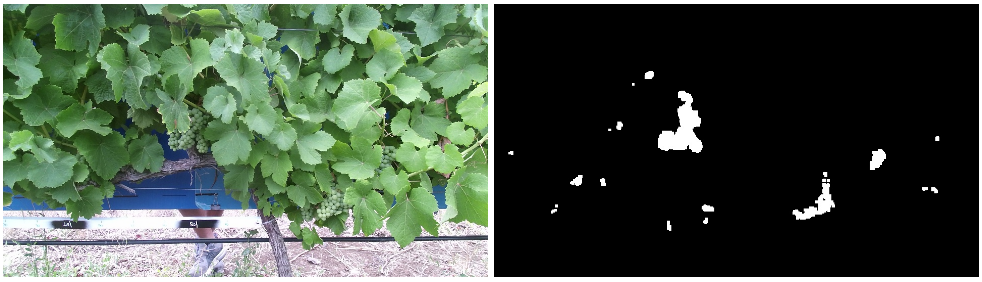


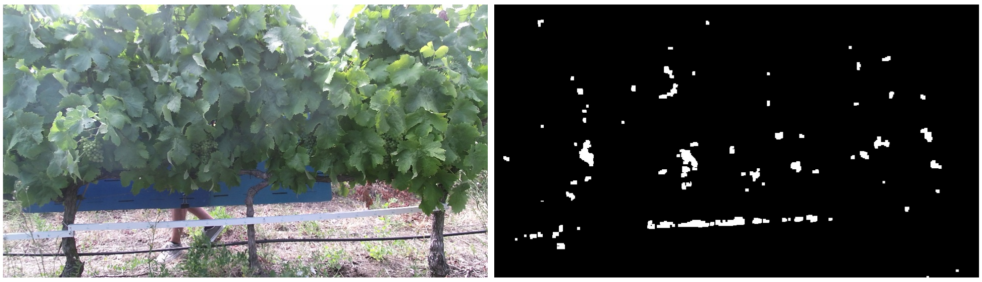
Figure 4.11: Meter by meter analysis of the results for the Arinto variety in the veraison stage for the automatic segmentation



(a) Original image

(b) Resulting mask

Figure 4.12: Example of an image of the Encruzado variety in veraison stage



(a) Original image

(b) Resulting mask

Figure 4.13: Example of an image of the Arinto variety in veraison stage

4.3.2 Harvest

Although the most useful estimation is made around three months before harvest in the veraison stage, these results can also be useful. The summed results that make the final yield estimation are given in Table 4.4, where the Encruzado estimation had an error of 107% and the Arinto had an error of 70%. Both these large errors point to segmentation failure. The manual estimation, as it would be expected, improved slightly when compared to the error from the veraison stage. Contrary to this, the automatic segmentation failed in these conditions. Several explanations are proposed in this chapter. The light source and illumination issues are discussed in detail and also a justification based also on the porosity and actual yield values is provided. The results for each individual meter, by order of SP, are shown in Figures 4.14 and 4.15.

Table 4.4: Final results for the yield estimation in kg in harvest stage with the associated relative error to the actual yield between parentheses

	Encruzado	Arinto
Actual yield	69,2 kg	123,4 kg
Hand segmentation	68,6 kg(1%)	122,5 kg(1%)
Automatic segmentation	143,50 kg(107%)	209,68 kg(70%)

It is noticeable in Figure 4.14 an overall overestimation of all meters, with a few exceptions. The meters with a more overwhelming error, such as 4, 17 and 22, were used to study the conditions and implications of these miss estimations. The conclusions are detailed towards the end of this chapter.

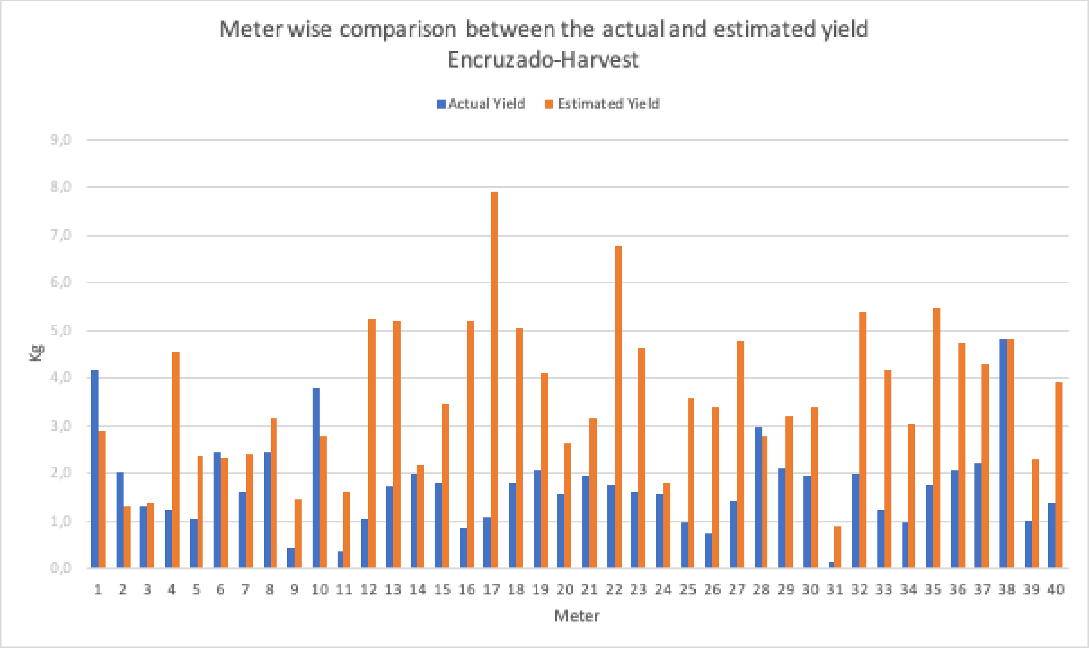


Figure 4.14: Meter by meter analysis of the results for the Encruzado variety in the harvest stage for the automatic segmentation

The Arinto results had the utility to see if the presumed causes for the miss estimation in the Encruzado set could be generalised to Arinto's worst cases, such as meter 3, 18 and 36.

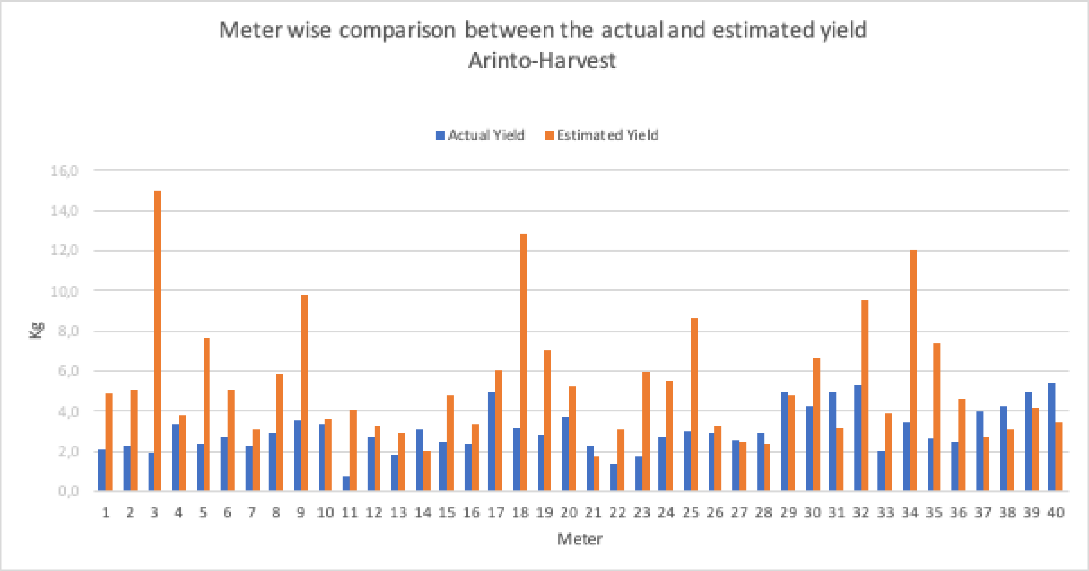


Figure 4.15: Meter by meter analysis of the results for the Arinto variety in the harvest stage for the automatic segmentation

With the data from the 2019 data set, it is possible to understand the importance of the data collection conditions, for them to be consistent and aware of environmental factors such as illumination. Also, the fact that the process is not fully automated may introduce some human error.

For both the Encruzado and Arinto the results were not satisfactory, especially considering that the manual segmentation produced equally good results as before when estimating in the veraison stage. The automatic segmentation estimated double the actual yield and the meter by meter error was also significantly off from the ground truth. This phenomenon has several possible explanations.

Firstly, these were not the expected results, especially taking into consideration the veraison estimations. The illumination conditions were a significant factor, along with other characteristics, in the segmentation errors. Starting with the Encruzado set, when compared to the veraison images, there are some noticeable differences. The light source relative position, for example. Although the mean light intensity in both sets is similar, the standard deviation is not for some of the images. Taking as an example the first SP in both sets, here are in Table 4.5 the standard deviations in 3 different representations, all normalised to 255, CIE LAB,Hue Saturation Value (HSV) and grey scale:

The average standard deviation in all tables varies in approximately 20 points from veraison to harvest, almost 10%. Although there is not a generalisation for some of the other SP, in Arinto or Encruzado, it is a factor that cannot be dismissed. This indicates that the network may not be robust enough to handle significant differences in illumination conditions, even though part of the pre processing was design to do just so. Other than the light distribution in the image, another issue is the light source and its

Table 4.5: Standard deviation for light measures for the first SP in both veraison and harvest data sets of Encruzado before splitting the image frames into the respective meters

(a) Standard deviation of L measures in CIE LAB normalised to 255 for the first SP of the Encruzado harvest and veraison sets

Frame	Veraison	Harvest
0	56.15	77.62
1	57.37	79.47
2	58.08	81.40
3	58.16	82.79
4	62.26	81.98
5	63.31	82.08
6	62.11	81.96
7	57.28	83.88
8	55.76	82.57
9	54.97	82.05
10	56.20	81.28

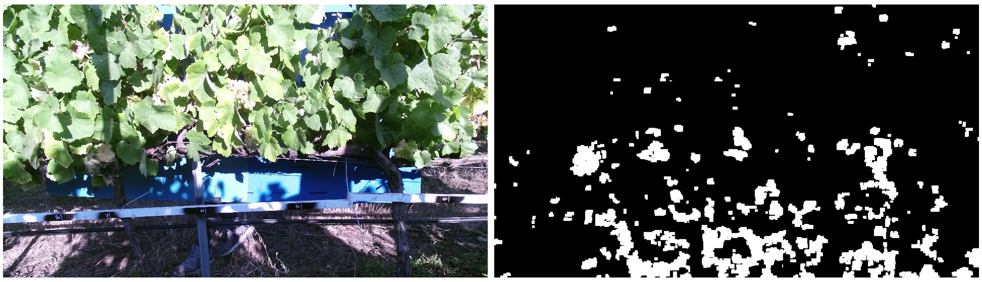
(b) Standard deviation of V measures in HSV colour space normalised to 255 for the first SP of the Encruzado harvest and veraison sets

Frame	Veraison	Harvest
0	56.35	78.61
1	57.35	79.45
2	57.60	81.48
3	58.05	82.85
4	62.09	82.29
5	62.90	82.51
6	61.62	81.90
7	57.45	83.36
8	55.31	82.07
9	54.98	81.58
10	56.50	81.24

(c) Standard deviation of light intensity in grey scale for the first SP of the Encruzado harvest and veraison data sets

Frame	Veraison	Harvest
0	58.26	78.90
1	60.36	80.44
2	60.45	81.37
3	60.56	80.76
4	64.67	80.80
5	66.14	81.08
6	64.99	83.48
7	59.63	81.39
8	58.62	81.52
9	57.28	80.47
10	59.04	79.48

position. Until the start of this work, the data collection was made with no specific rules regarding illumination. It was made without taking into consideration that different times of day and positions relative to the robot may interfere in the automatic segmentation. For example, as it is possible to see in Figures 4.16 and 4.17, a focus of light that comes from behind the robot causes shadows on the ground and canopy, also creating focus of illumination on the canopy that may induce in error even when performing hand segmentation.



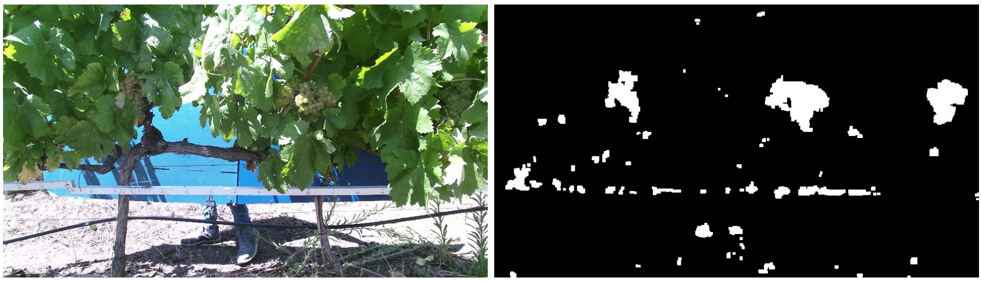
(a) Original image

(b) Resulting mask

Figure 4.16: Example of an image of the Encruzado variety in harvest stage with corresponding segmentation

This is not the case for the veraison data, since it was taken either with the light source behind the canopy, as it is possible to see in Figure 4.13a, or on a cloudy day where the illumination is constant from any side, as is in Figure 4.12a. These illumination problems were not supposed to be significant, since part of the pre processing was to change the colour space of the images, to reduce the effect of light, normalising the colour.

Other than the illumination issue, another correlation that was found was between the porosity values, the yield's ground truth and the absolute error in kg. Looking at Figures 4.14 and 4.15, even though there



(a) Original image

(b) Resulting mask

Figure 4.17: Example of an image of the Arinto variety in harvest stage with corresponding segmentation

is a general overestimation ate each SP, there are specific meters that stand out. There are two factors that most of these points share, a low porosity percentage and low actual yield. A low porosity is defined here as bellow the median. The median was chosen to threshold since the values in study were either relatively lower or higher than the mean, which was not representative of the sample. Since the network is prone to have false positives, any overestimation caused by the segmentation, which in turn was influenced by the data set's conditions, would be amplified by the model, overestimating the yield based on the overestimation of visible grape bunches. For example, in meter 17 of the Encruzado harvest set(Figure4.14), the absolute error is of 6.8kg. There was an overestimation of visible bunches in the automatic segmentation. Also, the porosity for this meter was of 2.5%, when the median is 4.5% and the ground truth was 1.1kg. For the Arinto harvest(Figure 4.15) there is also other examples such as meter 3, where the absolute error is of 13 kg and the porosity was just 2.7%. When considering the images with low porosity they represent almost 60kg of error in the Arinto harvest set and approximately 45kg in the Encruzado harvest set. These type of errors also occur in the veraison stage, but since the data set and the training set are alike, unlike the harvest set, there is a more accurate segmentation and less noticeable errors.

All the aforementioned factors have relevance in the quality of the segmentation and, consequently, in the yield estimation. None is responsible by itself, but a combination of them explain the results obtained from these 2 data sets. In general, the main conclusion that can be taken from these experiments is that the network is not robust enough to handle conditions that are significantly different from the ones it was trained on, because the training set is not diverse enough to be representative of those different conditions.

4.4 Summary

In this chapter the several experiments made were explained in further detail. Starting with the experiments performed relative only to the segmentation machine, a naive baseline was defined and from there the experiments were made, always adding to the system. After the hyper parameters were decided, followed the experiments relative to the yield estimation, with a different data set. These experiments allowed for a deeper understanding of the robustness of the algorithm, recognising where it fails and succeeds and why in both cases. The data collection step showed to be a very important one in the project. Overall, the illumination conditions in the data sets and the lack of diversity of the training set were the main variables in determining the quality of the segmentation. Aspects from the illumination standard deviation to others that are not quantifiable, such as the light source position, were studied to better understand how they interfere with the segmentation and consequent estimation. Other than the illumination, the porosity percentage and ground truth yield showed to influence the estimation error, aggravating any miss classification that may have happened before. In Chapter 5, suggestions to correct these fragilities in the system are presented.

5

Conclusion and future work

Contents

In this chapter, the final conclusions are presented together with an overall summary of the entire work and results. Some possible improvements to the project are further explained, some of them were already mentioned in previous chapters.

5.1 Conclusions

To address the main goal of this dissertation, a system described in Chapter 3 was proposed and presented. It was comprised of different parts, the main one for this work being the automatic segmentation. The focus was replacing the hand made segmentation that was performed until the development of this dissertation. This part, isolated from the others of the project, was tested achieving satisfying results of up to 64% of Intersection Over Union (IOU) in the test set. With a Fully Convolutional Network (FCN) trained only with limited data collected from the Instituto Superior de Agronomia (ISA) vineyards, the segmentation matched the man made in most of the cases. This testing resulted in the adding of other components in the process, namely a pre and post processing, that were essential for the score of 64%. Both these parts bettered the segmentation metric scores until that point. The pre processing consisted of applying a sliding window to the image for formatting purposes, performing data augmentation and converting the original colour space, Red-Green-Blue (RGB) into Commission internationale de l'éclairage Lightness* a* b* (CIE LAB). The data augmentation performed achieved positive results, as did the colour space transformation. As for the post processing, the morphological operation "open" was used, reducing the main problem of false positives in the image by eliminating small false positive clusters and by smoothing the grape cluster boundaries. This problem was also addressed by the reconstruction of the image, that was made in such a way that the overlapping areas were used to reduce false positives as well. This part of the experiments confirmed that it is possible to train a FCN end-to-end with a limited data set for a segmentation problem with an imbalance data set, which by itself already is a sub problem to the main one.

After obtaining reasonable results in the segmentation evaluation, the final test is to use the models used on the hand made segmentation to estimate the yield for the same areas, the 40 meters of Smart Points (SP). The images belonging to this different data set were passed through the system, resulting in the segmentation masks for each image. These masks had to be cropped to correspond to the meters that they contained in the image, so that a meter by meter comparison could be made with the actual yield.

There are benefits to have a prediction made ate every stage of the grape's development, and the earlier the better. The prediction made one month before, at the veraison stage, being accurate, it is very useful. Experiments were made for both the veraison stage and the harvest stage, for two varieties, Encruzado, that was used for training, and Arinto. The results for the veraison stage were significantly more promising than the harvest ones, with relative errors to the actual yield in the accepted interval of 0 to 10%. Firstly, the results help to make the point relative to the data collection, since they are very similar between the varieties but different between stages. Secondly, the results show that it is possible to successfully replace hand segmentation by an autonomous one, in certain conditions, and, with future work those conditions may be broader. The harvest results, although not satisfactory, were useful to learn the importance of correct and consistent data collection and to expose fragilities in the segmentation network, more specifically, the inability to handle significantly different conditions to the ones present in the training data set, which can contribute to the point about the fragility of a neural network, the need of extensive data to perform correctly.

5.2 Future work: data collection and further training

One of the most important stages in this process is the data collection. As it was seen in Chapter 4.3, a lack of care can result in data that the network is not able to segment. The consistency of this part of the process is essential for good results. Therefore, there are two aspects that need to be improved in this context: data collection conditions and the diversity of the training set. Firstly, the illumination conditions for the future data sets should match the conditions in the training set, as much as it is possible to control, given that there are inevitabilities in nature beyond human control. Secondly, to handle discrepancies, the network should be retrained, this time with a more representative data set, possibly including the 2019 images when an appropriate ground truth is available. Other than this, data augmentation regarding illumination conditions should also be performed, since there is an associated cost to creating new data sets and corresponding ground truth and a more complete approach to colour normalisation should be studied, in order to complement the pre processing and provide better invariance to these changes in conditions. Also, in the next years, the network should be retrained with new data in order to understand if there is a significant improvement with time variability. Another test that should be performed is the evaluation with different varieties, that should lead to a deeper comprehension of the fragilities of the network, which in turn provides useful information over which it is possible to decide if, for example, a better system for different varieties should be to train for each only with that variety's data.

As stated throughout this work, the network's training had limited access to data and to computational resources, for example, the training exhausted the google colab platform's time. With the use of a computer without time concerns it may be also possible to improve the results presented in this work. This additional training can take new inputs, as stated before. Also, the estimation models can also be adapted for the automatic segmentation, in a consistent way, producing better results.

Bibliography

[1] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 3431–3440.

[2] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” *CoRR*, vol. abs/1505.04597, 2015. [Online]. Available: http://arxiv.org/abs/1505.04597

[3] J. Fourie, J. Hsiao, and A. Werner, “Crop yield estimation using deep learning.” Zenodo, Oct. 2017. [Online]. Available: https://doi.org/10.5281/zenodo.893710

[4] R. Rudolph, K. Herzog, R. Töpfer, and V. Steinhage, “Efficient identification, localization and quantification of grapevine inflorescences in unprepared field images using fully convolutional networks,” 2018.

[5] E. Chavolla, D. Zaldivar, E. Cuevas, and M. Cisneros, *Color Spaces Advantages and Disadvantages in Image Color Clustering Segmentation*, 01 2018, pp. 3–22.

[6] V. Dumoulin and F. Visin, “A guide to convolution arithmetic for deep learning,” 2016.

[7] M. G. Selvaraj, A. Vergara, H. Ruiz, N. Safari, S. Elayabalan, W. Ocimati, and G. Blomme, “AI-powered banana diseases and pest detection,” *Plant Methods*, vol. 15, no. 1, p. 92, 2019. [Online]. Available: https://doi.org/10.1186/s13007-019-0475-z

[8] R. Szeliski. (2011) Computer vision algorithms and applications. London; New York. [Online]. Available: http://dx.doi.org/10.1007/978-1-84882-935-0

[9] J. Queiroz, “Estimativa da produção de uva na casta Encruzado com recurso a análise de imagem,” Master’s thesis, Instituto Superior de Agronomia, 2018.

[10] M. Bergerman, J. Billingsley, J. Reid, and E. van Henten, *Robotics in Agriculture and Forestry*. Cham: Springer International Publishing, 2016, pp. 1463–1492. [Online]. Available: https://doi.org/10.1007/978-3-319-32552-1_56

[11] P. R. Clingeleffer, S. R. Martin, G. M. Dunn, and M. P. Krstic, *Crop development, crop estimation and crop control to secure quality and production of major wine grape varieties : a national approach : final report to Grape and Wine Research & Development Corporation*, S. Martin and G. Dunn, Eds. Adelaide, Australia: Grape and Wine Research & Development Corporation, 2001.

[12] R. BRAMLEY and R. HAMILTON, "Understanding variability in winegrape production systems," *Australian Journal of Grape and Wine Research*, vol. 10, no. 1, pp. 32–45, 2004. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1755-0238.2004.tb00006.x

[13] G. Victorino, R. Braga, and C. M. Lopes, "The effect of topography on the spatial variability of grapevine vegetative and reproductive components," 2017.

[14] A. Milella, R. Marani, A. Petitti, and G. Reina, "In-field high throughput grapevine phenotyping with a consumer-grade depth camera," *Computers and Electronics in Agriculture*, vol. 156, pp. 293 – 306, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168169918307580

[15] B. Keresztes, F. Abdelghafour, D. Randriamanga, J.-P. Da Costa, and C. Germain, "Real-time Fruit Detection Using Deep Neural Networks," in *14th International Conference on Precision Agriculture*, Montréal, Canada, 2018. [Online]. Available: https://hal.archives-ouvertes.fr/hal-02518559

[16] G. M. DUNN and S. R. MARTIN, "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, no. 3, pp. 196–198, 2004. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1755-0238.2004.tb00022.x

[17] G. Victorino, G. Maia, J. Queiroz, R. Braga, J. Marques, J. Santos-Victor, and C. Lopes, "GRAPEVINE YIELD PREDICTION USING IMAGE ANALYSIS - IMPROVING THE ESTIMATION OF NON-VISIBLE BUNCHES G." Rhodes Island, Greece: 12th EFITA International Conference, 2019, pp. 60–65.

[18] S. Martin, R. Dunstone, and G. Dunn, "How to forecast wine grape deliveries," 10 2003, p. 100.

[19] M. Cunha, H. Ribeiro, and I. Abreu, "Pollen-based predictive modelling of wine production: Application to an arid region," *European Journal of Agronomy*, vol. 73, pp. 42–54, 2 2016.

[20] J. M. Tarara, B. Chaves, L. A. Sanchez, and N. K. Dokoozlian, "Use of cordon wire tension for static and dynamic prediction of grapevine yield," *American Journal of Enology and Viticulture*, 2014.

[21] M. Reis, R. Morais, E. Peres, C. Pereira, O. Contente, S. Soares, A. Valente, J. Baptista, P. Ferreira, and J. Bulas Cruz, "Automatic detection of bunches of grapes in natural environment from color images," *Journal of Applied Logic*, vol. 10, no. 4, pp. 285 – 290, 2012, selected papers from the 6th

International Conference on Soft Computing Models in Industrial and Environmental Applications. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1570868312000535

[22] S. Liu, S. Cossell, J. Tang, G. Dunn, and M. Whitty, “A computer vision system for early stage grape yield estimation based on shoot detection,” *Computers and Electronics in Agriculture*, vol. 137, pp. 88 – 101, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168169916311334

[23] B. Millan, A. Aquino, M. P. Diago, and J. Tardaguila, “Image analysis-based modelling for flower number estimation in grapevine,” *Journal of the Science of Food and Agriculture*, vol. 97, no. 3, pp. 784–792, 2017. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/jsfa.7797

[24] S. Liu, X. Li, H. Wu, B. Xin, J. Tang, P. R. Petrie, and M. Whitty, “A robust automated flower estimation system for grape vines,” *Biosystems Engineering*, vol. 172, pp. 110 – 123, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1537511017304610

[25] S. Nuske, K. Wilshusen, S. Achar, L. Yoder, S. Narasimhan, and S. Singh, “Automated visual yield estimation in vineyards,” *Journal of Field Robotics*, vol. 31, no. 5, pp. 837–860, 2014. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21541

[26] A. Aquino, M. P. Diago, B. Millán, and J. Tardáguila, “A new methodology for estimating the grapevine-berry number per cluster using image analysis,” *Biosystems Engineering*, vol. 156, pp. 80 – 95, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1537511016300940

[27] N. Atif, M. Bhuyan, and S. Ahamed, “A review on semantic segmentation from a modern perspective,” in *2019 International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, 2019, pp. 1–6.

[28] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. Garcia-Rodriguez, “A review on deep learning techniques applied to semantic segmentation,” 2017.

[29] Y. Guo, Y. Liu, T. Georgiou, and M. S. Lew, “A review of semantic segmentation using deep neural networks,” *International Journal of Multimedia Information Retrieval*, vol. 7, no. 2, pp. 87–93, 2018. [Online]. Available: https://doi.org/10.1007/s13735-017-0141-z

[30] R. Pérez-Zavala, M. Torres-Torriti, F. A. Cheein, and G. Troni, “A pattern recognition strategy for visual grape bunch detection in vineyards,” *Computers and Electronics in Agriculture*, vol. 151, no. May, pp. 136–149, 2018. [Online]. Available: https://doi.org/10.1016/j.compag.2018.05.019

[31] W. Lee, R. Ehsani, J. Schueller, V. Alchanatis, and H. Gan, “Immature green citrus fruit detection using color and thermal images,” *Computers and Electronics in Agriculture*, vol. 152, no. July, pp. 117–125, 2018. [Online]. Available: https://doi.org/10.1016/j.compag.2018.07.011

[32] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the Inception Architecture for Computer Vision,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-Decem, pp. 2818–2826, 2016.

[33] A. Krizhevsky, I. Sutskever, and G. Hinton, “Imagenet classification with deep convolutional neural networks,” *Neural Information Processing Systems*, vol. 25, 01 2012.

[34] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” pp. 1–14, 2014. [Online]. Available: http://arxiv.org/abs/1409.1556

[35] “Cielab colour space,” https://sensing.konicaminolta.asia/what-is-cie-1976-lab-color-space/, accessed: 2020-08-13.

[36] P. Rosero, D. Peluffo, A. Umaquinga, E. Rosero, and D. Peña, “Data visualization using interactive dimensionality reduction and improved color-based interaction model,” vol. 10338, 05 2017.

[37] J. Brownlee, *Deep Learning for Computer Vision: Image Classification, Object Detection, and Face Recognition in Python*. Machine Learning Mastery, 2019. [Online]. Available: https://books.google.pt/books?id=DOamDwAAQBAJ

[38] A. Saravanan, G. Perichetla, and D. K. S. Gayathri, “Facial emotion recognition using convolutional neural networks,” 2019.

[39] S. B. Kotsiantis, D. Kanellopoulos, and P. E. Pintelas, “Handling imbalanced datasets: A review,” 2006.

[40] R. Shi, K. N. Ngan, and S. Li, “Jaccard index compensation for object segmentation evaluation,” in *2014 IEEE International Conference on Image Processing (ICIP)*, 2014, pp. 4457–4461.

[41] Y. Yuan and Y. Lo, “Improving dermoscopic image segmentation with enhanced convolutional-deconvolutional networks,” *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 2, pp. 519–526, 2019.

[42] R. Bonaria, “Grapevine yield estimation using image analysis for the variety arinto,” Master’s thesis, Instituto Superior de Agronomia, 2019.

[43] C. Lopes, J. Graça, J. Sastre, M. Reyes, R. Guzmán, and R. Braga, “Vinbot robot-preliminary results with the white variety viosinho,” 2016. [Online]. Available: http://www.vinbot.eu/

[44] C. Hacking, N. Poona, N. Manzan, and C. Poblete-Echeverría, “Investigating 2-d and 3-d proximal remote sensing techniques for vineyard yield estimation,” *Sensors*, vol. 19, 08 2019.

[45] B. Carmignani, “Comparison of different methodologies to estimate bunch compactness,” Master’s thesis, Instituto Superior de Agronomia, 2019.

[46] G. Victorino, “O efeito da posição topográfica no desenvolvimento , produtividade e qualidade em diferentes castas na vinha,” Master’s thesis, Instituto Superior de Agronomia, 2015.

[47] A. Monteiro, G. Teixeira, C. Santos, and C. M. Lopes, “Leaf morphoanatomy of four red grapevine cultivars grown under the same terroir,” *E3S Web of Conferences*, vol. 50, p. 01038, 2018.

[48] D. M. Kline and V. L. Berardi, “Revisiting squared-error and cross-entropy functions for training neural network classifiers,” *Neural Computing & Applications*, vol. 14, no. 4, pp. 310–318, 2005. [Online]. Available: https://doi.org/10.1007/s00521-005-0467-y

[49] E. Carrillo, A. Matese, J. Rousseau, and B. Tisseyre, “Use of multi-spectral airborne imagery to improve yield sampling in viticulture,” *Precision Agriculture*, vol. 17, 07 2015.