Article

A Machine-Learning Approach for Automatic Grape-Bunch Detection Based on Opponent Colors

Vittoria Bruni †, **Giulia Dominijanni** † and **Domenico Vitulano** *,†

Department of Basic and Applied Sciences for Engineering, Sapienza Rome University, Via Antonio Scarpa 16, 00161 Rome, Italy; vittoria.bruni@uniroma1.it (V.B.); giulia.dominijanni@uniroma1.it (G.D.)
† These authors contributed equally to this work.

Abstract: This paper presents a novel and automatic artificial-intelligence (AI) method for grape-bunch detection from RGB images. It mainly consists of a cascade of support vector machine (SVM)-based classifiers that rely on visual contrast-based features that, in turn, are defined according to grape bunch color visual perception. Due to some principles of opponent color theory and proper visual contrast measures, a precise estimate of grape bunches is achieved. Extensive experimental results show that the proposed method is able to accurately segment grapes even in uncontrolled acquisition conditions and with limited computational load. Finally, such an approach requires a very small number of training samples, making it appropriate for onsite and real-time applications that are implementable on smart devices, usable and even set up by winemakers.

Keywords: bunch detection; color image processing; opponent colors; human perception; support vector machine (SVM)

Citation: Bruni, V.; Dominijanni, G.; Vitulano, D. A Machine-Learning Approach for Automatic Grape-Bunch Detection Based on Opponent Colors. *Journal Not Specified* **2023**, *1*, 0. https://doi.org/

Academic Editor: Firstname Lastname

Received:
Revised:
Accepted:
Published:

1. Introduction

The problem of a precise yield estimation in vineyards is of great interest for wine industry. Some data, such as a production of 6.04 million tons in 2020 only in the USA or a savings of about one hundred million dollars with a correct yield prediction, help to understand the importance of this topic [1,2]. As a result, increasing research work has been done in recent years on this topic through adopting different strategies. However, a precise yield prediction is as simple in theory as it is difficult in practice. Usually this task is accomplished by winemakers with unavoidable errors due to different factors that can be:

- *Objective:* non vineyard uniformity, weather conditions, different pruning techniques, etc. [3].
- *Subjective:* human overestimation, lack of attention, errors, etc. [4].

That is why an automatic image-based framework that replicates winemakers inspections but it is robust to external conditions, is becoming of great interest for the entire research field.

Objective factors make the task very difficult for any image-based framework exploiting deep-learning networks. In fact, the non uniformity of vineyards makes such an estimate complicated and strongly case dependent. For example, different kinds of pruning can amplify problems, such as object (i.e., grape) occlusions and so on. This difficulty is proven by the quantity of different kinds of deep neural networks (DNNs) proposed in the literature—see, for instance, [5,6] and the next section for a short review. Moreover, DNNs have a discrete computational burden (even though some approaches dealing with this problem have recently been proposed) [7–11] and need for a large and representative training set to guarantee an acceptable accuracy rate and to avoid overfitting [12]. Some recent approaches [13,14] have attempted to exploit pretrained convolutional neural networks (CNNs) to overcome their computational burden.

However, limits in finding useful images for an effective training for 'in-the-wild' cases along with the need for an RGB-d camera in place of a common RGB one prove that the grape bunch segmentation problem is far from being solved.

The goal of the proposed approach is to start from the aforementioned problems to produce a framework that:

- Is automatic, or at least, minimizes any human aid, while remaining effective for a reliable yield assessment.
- Is not computationally expensive, i.e., it allows for a fast response for each image and requires simple operations that facilitate its implementation on portable instrumentation.
- Requires a small training set, allowing its straightforward updating and adaptation to different conditions and use cases.

The aforementioned requirements come from the analysis of two possible practical scenarios. The first one accounts for unmanned aerial vehicle (UAV)-based applications where no web connections are available—very frequent in many practical cases. In this case, the software should run on the (small) computer that the UAV is equipped with. Hence, a simple artificial-intelligence (AI) tool that needs a low computational effort is required. The second scenario is the one where a winemaker uses their own smartphone for training and testing in the simplest way. In this case, very few examples for training the adopted AI tool are required, apart from a small computational effort. Both scenarios lead to a light and case-dependent approach based on a light machine-learning method (see Figure 1) [12]—with limited but good examples for the training set.

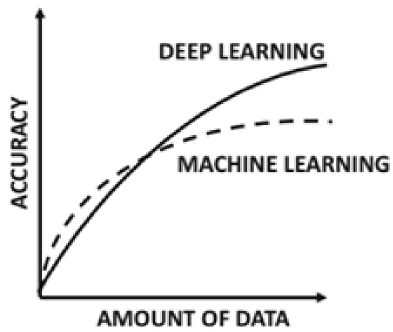

Figure 1. Number of data for training versus classification accuracy: comparison between machine-learning (ML) and deep-learning (DL) methods [12].

In addition to the considerations above, the subjective component plays a fundamental role. In fact, if on the one hand, the human approach has the drawback that winemakers 'tend to choose healthier and larger bunches when they are doing sampling in the field' [4], it is also true that humans have an undoubtable ability in recognizing objects in very different and critical conditions: that is why DNNs attempt to simulate it. Only the good part of human activity, its early visual perception, should be accounted for [15].

The proposed approach attempts to embed all aforementioned requests. It is based on the main peculiarities of human perception exploiting both the early vision processes in terms of luminance and contrast and the opponent colors theory. The latter has been formalized in the past years and is currently under investigation [15–19]. Specifically, the proposed method consists of a supervised learning framework oriented to identify the areas containing (yellow or blue) bunches of grapes in an 'in-the-wild' RGB vineyard image. With 'in the wild', we mean an image containing grape bunches as well as foliage, ground, and sky and in uncontrolled light conditions. The proposed method takes advantage of the use of limited but distinctive features that are close to the ones that are encoded in the onsite visual inspection process.

Exploiting both multiscale analysis and opponent colors theory, a new feature space is defined where each transformed image is analyzed by a suitably trained support vector machine (SVM). The rationale is to exploit the fact that human perception works as an optimized encoder for processing and storing visual information [15,20–22]. This property contributes to determining the right features for very effective bunch detection. The achieved results show that the proposed method is able to outperform competing approaches in terms of accuracy, size of the training set and computing time.

2. Materials and Methods

This section is organized as follows. First, a short review on available approaches in the literature is presented. This helps the reader to better understand the main guidelines followed in the literature. Successively, a technical background useful to understand the proposed approach is offered. It contains a sketch of both multiscale analysis and opponent colors theory. Finally, the proposed approach is described, outlining the peculiarities of blue and yellow grape cases.

2.1. A Short Review on Automatic Yield Estimation

The increasing interest on this topic makes it difficult for any review to be exhaustive. In the following, only the approaches that are related (or involve topics close) to the proposed approach will be presented.

The approach in [23] is a good example to understand the difficulty of making any automatic yield estimation feasible: it is as effective as it is computationally demanding. If on the one hand, the use of a radial symmetry transform allows effective two-stage and large-scale wine images processing, on the other hand, the complexity is of about $O(KN)$, with K being the pixel number of the whole image while N is the size of the neighborhood [24]. The computational burden is not the only problem for automatic yield estimation. In fact, under the hypothesis of using an UAV (or any automatic vehicle) with a suitable camera, various problems, such as the view angle and size of the scene (in order to take grape bunches), calibration [25], image resolution, light conditions, and acquisition conditions (to avoid grape occlusion) are crucial for a successive but effective color image processing. All these problems make classical image processing infeasible on high-resolution images [1].

Once the acquisition phase has been made, the successive processing is, again, not simple, and different approaches have been proposed. Most of them are clearly oriented to feature extraction and classification, where a high accuracy (very often in ideal conditions) is combined with a high computational effort. An example is in [26], where different Fuzzy C-Means (FCM) clustering methods are compared: Robust Fuzzy Possibilistic C-Means, FCM and FCM-GK (FCM with Gustafson–Kessel)—with accuracy ranging from 85% to 88%. Other interesting approaches are in: [27], which employs a 3D grapevines formation based on Structure-From-Motion followed by a saliency map analysis and SVM for classification; [26,28], where a combination of SVM, K-means and the scale-invariant feature transform (SIFT) for various vineyard components clustering is used; and [29,30], which classifies different vineyard objects by means of Mahalanobis measures.

As correctly outlined in [1], most of the computational effort is spent on processing unuseful scene components—estimated in at least 50% of the total information.

Some interesting approaches dealing with color images are found in [25,31,32]. The first two papers use RGB thresholding along with some morphological operations that speed up the processing phase even under controlled light conditions (artificial light in the night) and ideal acquisition conditions (controlled grape pose) with a manual RGB key value selection for successive SVM classification. It is straightforward that such an approach cannot be used in practice where large scale acquisitions are needed. From this point of view, an effort was made in [25] where a preliminary selection of bunch areas was made via color thresholding and morphological operations. This allows for successive feature selection and classification achieved via ReliefF [33], a sequential feature selection method [34] and SVM [35]. The preliminary selection of potential bunch areas allows for at least 70% of (useless) scene information to be discarded.

It is also worth mentioning the great effort in exploiting deep-learning techniques, very often in combination with computer vision technologies [12,36], to automate agricultural processes. In particular, many techniques have been inherited by a well-known field of computer vision: object detection. In this case, the very critical conditions (very different outdoor light conditions, different scale of the target, occlusions, and so on) in agriculture make this task very difficult. That is why various deep-learning approaches, mainly based on convolutional neural networks (CNN), have been proposed [5,6]. Specifically, deep-learning approaches for object detection can be split into:

- *One-stage detectors:* In this case, object classification and bounding-box regression are done directly without using pre-generated region proposals (candidate object bounding-boxes). Approaches belonging to this class are, for example, Single Shot multibox Detector (SSD) [37], RetinaNet [38], Fully Convolutional One-Stage (FCOS) [39], DEtection TRansformer (DETR) [40], EfficientDet [41], and the You Only Look Once (YOLO) family [42–46].
- *Two-stage detectors:* First, a generation of region proposals, e.g., by selective search as in R-CNN and Fast R-CNN or by a Region Proposal Network (RPN) as in Faster R-CNN, is made. Then, a second step oriented to object classification in each region proposal is applied. Sometimes, some additional phases, such as bounding-box regression for refining the region proposals, and binary-mask prediction, are performed. Examples of approaches belonging to this class are: region-based CNN (R-CNN) [47], Fast/Faster R-CNN [48,49], Spatial Pyramid Pooling Networks (SPPNet) [33], Feature Pyramid Network (FPN) [50], and CenterNet2 [51].

Usually, two-stage detectors perform better than one-stage ones in terms of the precision of localizing target in different conditions (see, for instance, [52,53]). However, two-stage detectors pay the price of 'slow inference speed and high requirement of computational resources' for this specific field [2]. This is the reason why many approaches oriented their effort toward one-stage detectors: specifically, YOLO networks (see, for instance, [7–11] and [54–56] for specific applications to different kinds of fruit). Apart from the specific adopted strategies, it is worth noting that all these approaches, even though fast, show a high sensitivity to occlusion and should be combined with further computer vision tricks [57].

In particular, a trend of a certain success is represented by Swin-Transformer (hierarchical vision transformer network)-based approaches [57–62]. Despite the interesting philosophy on which they are based (self-attention mechanism for learning [63–65]), again, the computational effort is still high. Interesting approaches based on this strategy and concerning the agricultural field can be found in [66–68]. Finally, it should not be overlooked that the intensive use of deep-learning networks leads to the need for a great quantity of images for suitable training. That is why the need for populated and labeled databases is becoming very impelling—two very recent databases on grape bunches are described in [69,70].

2.2. Technical Background

This section focuses on two main topics that the proposed model is based on: multiscale analysis and the opponent colors theory. These will be the focus of the next two sections, respectively.

2.2.1. Multiresolution Analysis

The change of scale of the input image can be seen as simple application of the multiresolution analysis theory. The latter involves the formal definition of producing different scales of a given function that are correlated to each other by some mathematical properties [71]. This representation is oriented to highlight specific details of a given signal in agreement with the pioneering studies on multiresolution pyramids by Burt and Adelson [72] first and the formal construction of orthogonal wavelets later [73].

Coarsely speaking, a given function f at a resolution 2^{-j} can be seen as a (discrete) grid of samples where local function averages are considered—the size of the average domain is proportional to 2^j. A multiresolution approximation of f is composed of different and embedded grids. Very often, this operation becomes more intuitive by considering each one of these grids (say, at resolution 2^{-j}) as the orthogonal projection on the space $V_j \subset L^2(\mathbf{R})$ (This functional space contains functions with finite energy.). V_j includes all possible approximations at the resolution 2^{-j}. Hence, starting from a given function f, its approximation f_j at resolution 2^{-j} is the projection on the space V_j constrained to minimize the following quantity: $||f - f_j||_2$.

More details concerning this theory and its applications can be found in [71,73,74]. However, it is possible to say that the aforementioned theory paves the way to orthonormal wavelets, i.e., orthonormal bases. More specifically, the approximation of a given function f at the resolution 2^{-i} can be defined as the orthogonal projection $P_{V_j} f$ on V_j. In order to find such a projection, an orthonormal basis of V_j has to be looked for. Usually, this operation can be achieved by convolving f with a dilated and translated version of a scaling function Φ. It is possible to prove that, under suitable conditions, the family $\{\Phi_{j,n}\}_{n \in \mathbf{Z}}$, with j and n, respectively, the dilation and shifting parameters, is an orthonormal basis of V_j for all $j \in \mathbf{Z}$.

In the sequel, only one smoothed version of f will be used for each of the two phases of the model, and the Haar basis was selected as Φ [71]. The latter can be simply seen as an operator that computes local averages of f with a fixed window that depends on the resolution level j. It is worth stressing that this smoothing is consistent with human vision mechanisms in the pre-attentive phase where redundant and not perceived frequencies are discarded [75,76]. Smoothing irregular areas has also the advantage of making regions more homogeneous and, thus, enhancing them and actually increasing their visual saliency [77,78]. The selection of a proper level of resolution quantifies the amount of information that can be lost in the visual coding process as formally studied in [76,78].

2.2.2. Opponent Colors Theory

It is well-known that the *trichromatic theory* of color vision explains how human beings (their cells) detect blue, red, and green wavelengths. The combination of these three main colors allows perception of the whole visible spectrum [15]. However, the current understanding of color perception is more complicated. In particular, a key role is played by the *opponent colors theory*. This theory was developed by Ewald Hering, who based it on the observation that specific color combinations cannot be seen [79]. The latter is based on the human ability to perceive color that is mainly based on three receptor complexes: the red–green complex, the blue–yellow complex, and the black–white complex [17]—see Figure 2. As matter of fact, recently, the pairings above have been refined as blue–yellow, red–cyan, and green–magenta.

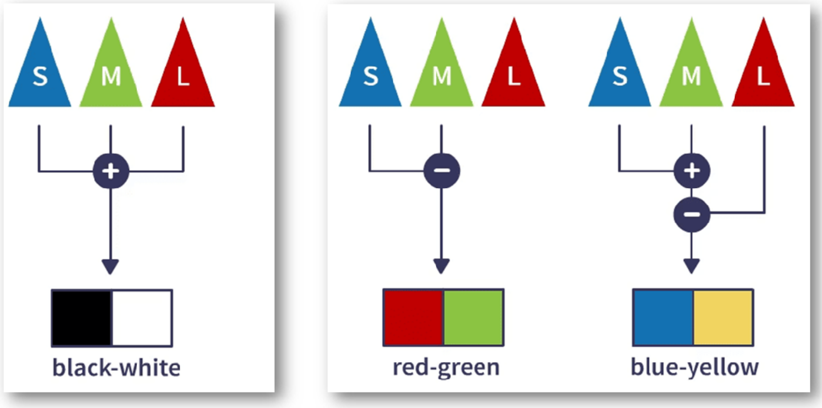

Figure 2. *Opponent process theory.* The three receptor complexes: black–white, red–green, and blue–yellow for (**Left**) an achromatic and (**Right**) chromatic case.

According to the opponent colors theory, human brain can only register the presence of one color of a pair at a time. Specifically, for each receptor complex, the involved colors oppose one another. For example, the cell that activates for red will deactivate for green light and vice versa. Two opponent colors cannot survive in vision. This mechanism can be seen as a perceptive (and very effective) coding of visual information.

The whole theory is then based on the presence of two kinds of (opposite) cells for each receptor complex, which activate for a color while having an inhibitory response to the opposite one [17].

Though not used in this paper, it is worth outlining how this theory can explain the perceptual phenomena of negative afterimages—see [15,17,80] and the references therein.

It is possible to say that, while the classical trichromatic theory helps to explain how different types of cones detect different light wavelengths, the opponent colors theory says how the cones connect to the ganglion cells and then how opposing cells are excited or inhibited by certain wavelengths of light. In addition to these two theories, there is the complementary color theory that accounts for how and which wavelengths translate to which colors and then how the brain processes these colors.

2.3. The Proposed Method

The proposed model is based on human perception as it attempts to exploit the ability of human beings in recognizing objects in a small amount of time. Specifically, the ability of human early vision will be exploited for our study case.

It is well-known that early vision refers to those stages of vision that involve capturing, preprocessing, and coding visual information but do not involve the interpretation or other cognitive processing of visual information that requires further brain processing [15]. The proposed model follows some recent results proving that the early vision phase accounts for visual information that is mainly based on the luminance and contrast of the scene under study [75]. In fact, luminance gain control (known as light adaptation) is managed in the retina and is oriented to adjust the sensitivity to match the locally prevalent luminance (light intensity).

Coarsely speaking, the retina divides luminance by the local mean luminance [81–83]. On the contrary, contrast gain control starts in the retina and is strengthened at some successive stages of the visual system [81,84–89]. Apart from specific details (that can be found in [90]), the input signal is divided by a measure that grows with the locally prevalent root-mean-square (r.m.s.) contrast. In this way, a contrast invariance is produced for a better processing of the eye response: a contrast increase when the contrast is low and vice versa.

In addition to the aforementioned 'visual normalization', it is worth highlighting another important aspect of visibility: the human eye works as a low pass filter at first glance [15,75]. This is due to the necessity of quickly understanding the content of the scene—for survival needs. That is why both luminance and contrast are considered at a given resolution in the proposed approach. This strategy has a double purpose: on the one hand, it replicates what happens in the early vision (the first 200 milliseconds) phase; on the other, it allows discarding a great deal of information that is not useful for the analysis of the scene content.

These two aspects are clearly linked each other and are implemented in a cascaded binary classification method in this paper. It is worth highlighting that the combination of more than one SVM classifier is not new in the literature, and it has been employed to improve classification accuracy. However, the optimization of a combination of more than one classifier is still matter of study for a wide part of the research community interested in effective classification tools—see, for instance, [91,92]. Figure 3 provides a graphical sketch of the proposed procedure that is described in the following.

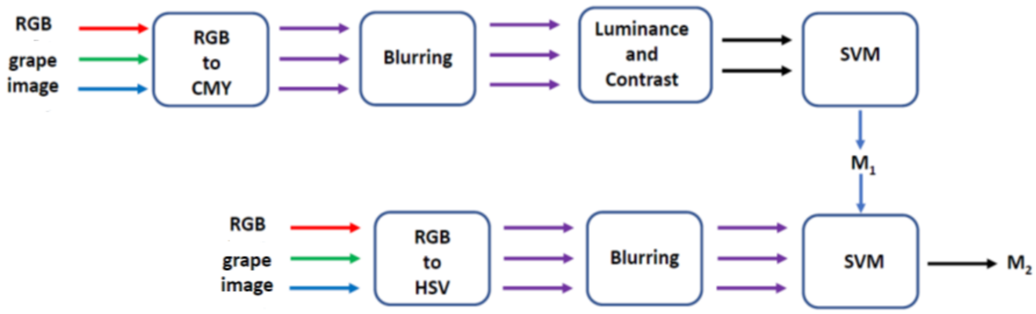

Figure 3. Block scheme of the proposed processing pipelines.

First classification. From an algorithmic point of view, an RGB image I depicting a sketch of a vineyard will be transformed into another color space that emphasizes the contrast between object (grape) and background (leaves, grass, sky, etc.). The selected color space is CMY (cyan, magenta, and yellow) as this allows us to emphasize the features of blue and yellow grapes as it will become clearer in the following.

The achieved projected image J in the new color space will be then convolved with a suitable kernel $\Phi_{\bar{j}}$ to simulate the low-pass filtering applied by the human visual system, thus, obtaining the image $\tilde{J}$. A multiscale analysis is then adopted where only one properly selected scale $(2^{\bar{j}})$ is considered. Hence, at each pixel location $\bar{x}, \bar{y}$ in the image spatial domain Ω, the luminance $L(\bar{x}, \bar{y})$ and the contrast $Con(\bar{x}, \bar{y})$ can be considered as components of the following vector of features

$$\mathbf{v}_1(\bar{x}, \bar{y}) = [L(\bar{x}, \bar{y}), Con(\bar{x}, \bar{y})] \quad \forall (\bar{x}, \bar{y}) \in \Omega. \tag{1}$$

By denoting with $\Omega_G \subset \Omega$ the grape image region and using the feature vector $\mathbf{v}_1$, it is then possible to classify the input image by means of an SVM binary classifier that produces the first binary map M_1, defined as follows:

$$M_1(\bar{x}, \bar{y}) = \begin{cases} 1 & \text{if} \quad (\bar{x}, \bar{y}) \in \Omega_G \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

The first phase accounts for the early vision mechanism on a (vineyard) color image. However, the role of the colors themselves is crucial in our problem as discriminant for grape recognition. That is why a second phase that refines the map M_1 is required. It accounts for the pure color information in order to discard non-grape pixels in M_1.

Second classification. For the second classification, the hue, saturation, and brightness components of the HSB space were used as features. HSB space (hue, saturation, and brightness) is also known as HSV (hue, saturation, and value). This space was introduced in the 1970s in order to better fit the way human vision perceives color-making attributes. Moreover, its definition is intuitive and, though debated, its cylindrical geometry (along with the HSL space) makes color perception more natural from a human point of view—see [15,93] for its formal and geometrical derivation.

Let K be the RGB image I projected in this color space; again, all color components are blurred in order to simulate the low-pass filtering performed by the naked eye. By denoting with $\tilde{K}$ the blurred image, for each $(\bar{x}, \bar{y}) \in \Omega_G$, the feature vector $\mathbf{v}_2$ is defined as

$$\mathbf{v}_2(\bar{x}, \bar{y}) = [\tilde{K}(\bar{x}, \bar{y}, 1), \tilde{K}(\bar{x}, \bar{y}, 2), \tilde{K}(\bar{x}, \bar{y}, 3)] \quad \forall (\bar{x}, \bar{y}) \in \Omega_G, \tag{3}$$

where the indices $1, 2, 3$, respectively, refer to the hue, saturation, and value components of $\tilde{K}$ and are used as input for a second SVM binary classifier. The SVM-based classification restricted to the non-zero entries of M_1 provides the binary map M_2 defined as follows,

$$M_2(\bar{x}, \bar{y}) = \begin{cases} 1 & \text{if} \quad M_1(\bar{x}, \bar{y}) = 1 \quad \text{and} \quad (\bar{x}, \bar{y}) \in \Omega_G \\ 0 & \text{otherwise,} \end{cases} \tag{4}$$

which represents the output of the proposed procedure.

To properly assess if an image pixel depicts part of a grape, the scheme described above has to be adapted to blue and yellow grapes through proper definitions of luminance and contrast as discussed in the following subsections.

2.3.1. Blue Grapes

Blue grape detection inherits the mechanism that characterizes the opponent color theory. In particular, the color of a ripe bunch is usually blue, and the intensity depends on the kind of vineyard and on the lighting conditions. On the other hand, the grape background is composed of leaves that have a certain shade of yellow—a reddish-yellowish. This is the reason why the selected color space is CMY: it contains both a shade of blue (cyan) and yellow. Keeping in mind the opponent color mechanism and its inhibitory action, we propose to code the above mechanism in terms of classical Weber contrast [15]. As a result, the contrast for the blurred image $\tilde{J}$ in the CMY color space is defined as

$$Con_b(\bar{x}, \bar{y}) = \frac{C(\bar{x}, \bar{y}) - Y(\bar{x}, \bar{y})}{Y(\bar{x}, \bar{y})}, \quad \forall\, (\bar{x}, \bar{y}) \in \Omega. \tag{5}$$

In addition, bearing in mind that the human eye is designed to see luminance and contrast, the Y component is selected as luminance:

$$L_b(\bar{x}, \bar{y}) = Y(\bar{x}, \bar{y}), \quad \forall\, (\bar{x}, \bar{y}) \in \Omega. \tag{6}$$

The rationale behind this choice is that the prevalent color detected by the human eye (while checking the yield) is the yellow color of the leaves—more than the blue grapes. It is worth outlining that the classical and more simple Weber contrast was selected in place of the Michelson one [15]. The motivation of such a choice is two-fold. It is simpler to use and characterize the relative opponent action of the two involved colors; color bands are smoothed with the aim of producing a uniform object on a uniform background as is the case in early vision.

That is why the role of contrast, as in the Michelson one, vanishes. In addition, even though there exist several studies concerning contrast in color images containing text, to the best of the authors' knowledge, there is not an explicit formula for color contrast, and the matter is still debated (In this case, the minimum contrast ratio should be 4.5:1 for normal text and 3:1 for large text. Various software tools to check this [94] are also available.).

Using Equations (5) and (6), for blue grapes, Equation (1) becomes

$$\mathbf{v}_1(\bar{x}, \bar{y}) = [L_b(\bar{x}, \bar{y}), Con_b(\bar{x}, \bar{y})], \quad \forall\, (\bar{x}, \bar{y}) \in \Omega.$$

It represents the input feature vector for the first binary SVM classifier producing the first map M_1.

Regarding the second classification step, as already mentioned in the previous section, it is necessary to also account for the peculiar grape color that changes in agreement with both the stage of grape ripening and the kind of vineyard. Hence, the feature vector $\mathbf{v}_2$ is defined as in Equation (3) and feeds the SVM-based classifiers producing the final classification map M_2.

2.3.2. Yellow Grapes

Though a slight modification, yellow grape detection follows the same strategy adopted for blue grapes. Before showing it, two considerations have to be made. The first is that yellow grape detection is more difficult. As shown in Figure 4 (Right), grapes have a color that is a mixture of red and yellow, while leaves are characterized by a mixture of green and yellow. The contrast in this case is more light. The second consideration is relative to the selected CMY space, where the subtractive primaries of cyan, magenta, and yellow are the opposing colors to red, green, and blue. Specifically:

- Cyan is opposite to red.
- Magenta is opposite to green.
- Yellow is opposite to blue.

Figure 4. Two examples (3264 × 2448 × 3 RGB images) of grapes in the considered vineyard: (**Left**) blue grapes and (**Right**) yellow grapes.

The following strategy was adopted to exploit the opposite components. In particular, magenta was selected as opposite to green (i.e., an approximation of leaves color), while cyan was selected as opposite to red (i.e., an approximation of grapes color). In this case, the common shade of yellow that characterizes both object (grape) and background (leaves) was considered to be self-vanishing.

The corresponding contrast and luminance, computed over the blurred color components, is:

$$Con_y(\bar{x},\bar{y}) = \frac{M(\bar{x},\bar{y}) - C(\bar{x},\bar{y})}{C(\bar{x},\bar{y})}, \quad \forall\,(\bar{x},\bar{y}) \in \Omega \tag{7}$$

and

$$L_y(\bar{x},\bar{y}) = C(\bar{x},\bar{y}), \quad \forall\,(\bar{x},\bar{y}) \in \Omega, \tag{8}$$

and then the feature vector $\mathbf{v}_1(\bar{x},\bar{y}) = [Con_y(\bar{x},\bar{y}), L_y(\bar{x},\bar{y})]$ is used as input for the first classifier. On the contrary, the second classifier works as for blue grapes.

3. Results

Experimental results and tests were performed in a vineyard located in Rome (San Cesareo), Italy in 2021. About 200 images were taken under natural light conditions. The adopted camera was a Kodak EasyShare V803. The image resolution was $3264 \times 2448 \times 3$. The vineyard was composed of different varieties of grape. In particular, Merlot, Cesanese, and Malvasia regarding blue grapes (oriented to wine production) and Uva Italia for yellow (table) grapes. The distance between the camera and grapes was about 2–3 m but was not expressly controlled. The same goes for the light conditions.

It is well-known in the literature that many available approaches have been tested in ideal conditions in terms of light, pose, without occlusions, and so on. This makes it very difficult to test a specific approach in real conditions [1]. Hence, in this paper, the choice of natural light conditions, no particular care to camera/grape distance as well as a vineyard with a type of pruning with many leaves was made in order to consider an 'in-the-wild' test. The proposed approach was tested on several images; the algorithm was run on a laptop (1.8 GHz Intel Core i5 dual-core, RAM 8 GB) in the MATLAB environment. Only some examples will be shown here, and they are the blue and yellow grape cases shown in Figure 4.

3.1. Blue Grapes

Following the steps in Figure 3, the input image in Figure 4 (Left) was converted into the complementary CMY color space.

Each component was then filtered by means of 2-D Haar filter $\Phi_{\bar{j}}$ with size 15×15. This choice is the simplest among wavelet filters to obtain a specific scale from a multiresolution analysis [71,78]. From these color components, both the luminance in Equation (6) and contrast in Equation (5), depicted in Figure 5, were considered as elements of the feature vector. The output of the first classification is the map M_1 shown in Figure 6. It is worth outlining that, in this case, the adopted training set was composed of only 50 (suitably selected) pixels, where the first 25 refer to 'grape' while the remaining 25 refer to 'other'—e.g., sky, soil, and foliage.

The second step of the proposed methodology requires the transformation of the RGB original image into the HSV color space. Each component has then been filtered by a Haar kernel of size 20×20. The size of the two adopted blurring kernels was tuned accounting for the maximum visual attention scale (usually the third or fourth for Haar kernels) in a wavelet decomposition [15,77,95]. Hence, for each pixel classified as 'grape' in M_1, the feature vector was built according to Equation (3).

The second SVM-based classification led to the M_2 map shown in Figure 7 (Left). The post-processing step consisted of a morphological opening [96] on the resulting binary map M_2, where the radius of the disk was set equal to 10. This step eliminates some spurious and isolated points due to a bad classification, thus, leading to the final map in Figure 7 (Right). This step has not been inserted in the scheme in Figure 3 as it simply refines the achieved result without greatly increasing the framework performance.

In particular, this step simply avoids some annoying and spurious points in the final map. The final classification is in Figure 8. As far it concerns the training set of the second classification, it was built by randomly selecting points among the 'good' ones in the first classification. Specifically, only points classified as grapes in the first classification were considered. Among them, the true grape points were used as 'grape', while the remaining ones represent the 'background'.

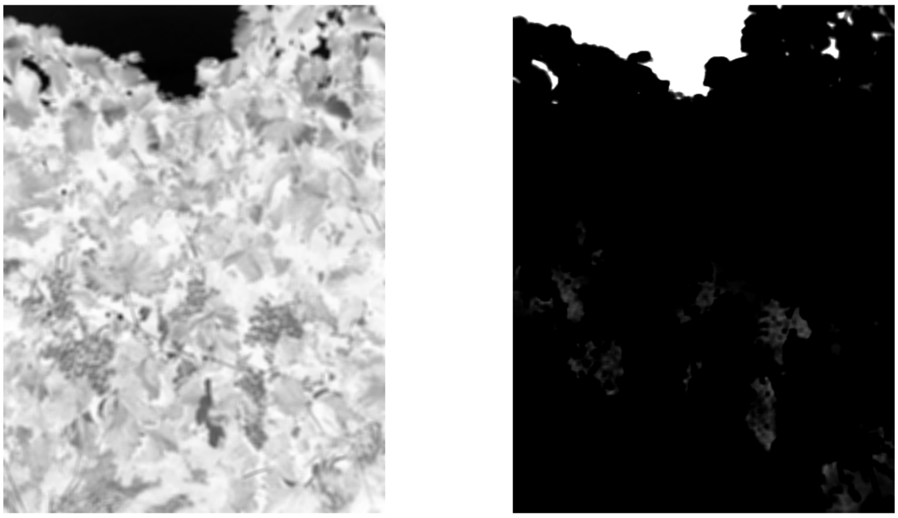

Figure 5. Blue grapes: (**Left**) luminance L_b as defined in Equation (6) and (**Right**) contrast Con_b as defined in Equation (5).

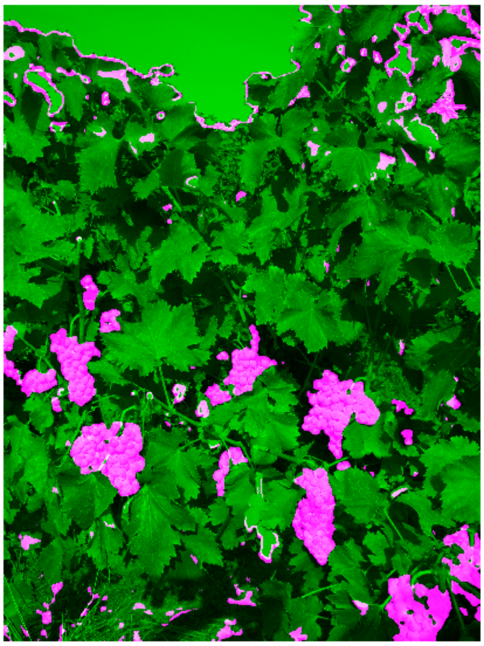

Figure 6. Blue grapes: M_1 map after the first classification superposed on the original RGB image—pixels classified as blue grapes are in magenta.

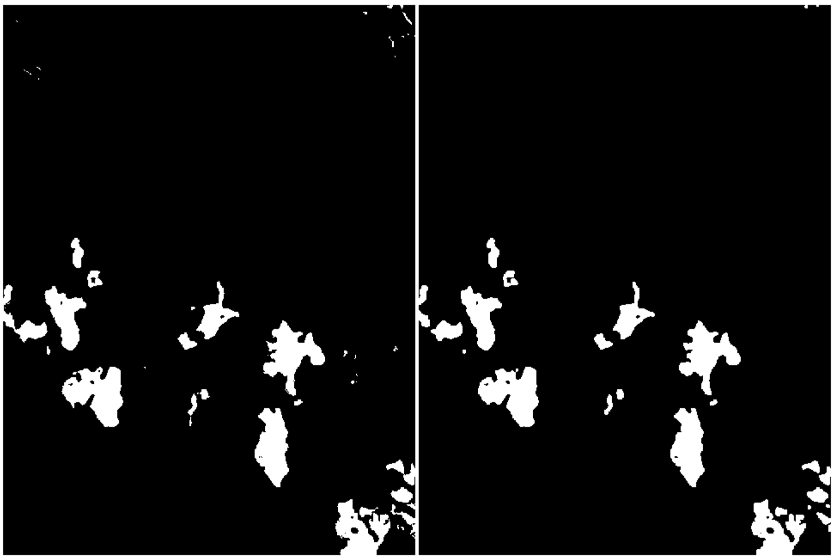

Figure 7. Blue grapes: (**Left**) M_2 map and (**Right**) its post-processed version.

The most interesting result of this paper is possibly contained in Table 1, where the classification accuracy is shown for both classifiers of the adopted cascade (first and second classification). The results are ordered in terms of increasing number of adopted points in the training set. As can be observed, only a few points are required for a correct classification. To further stress this point, Table 2 shows that having only 10 points in the training set can guarantee a classification accuracy greater than 95%.

This represents one of the main contributions of this work since it allows for a manual selection of points even by winemakers through a very fast procedure, making the proposed method easily and on-site adaptable to the different use cases and scenarios. Figure 9 shows the final map achieved using only 10 points, along with the corresponding classification accuracy, while Table 1 also contains the computing time. The latter refers to the main steps of the two cascaded classifications—the time required by the whole procedure is also provided. As can be observed, the whole process reaches high accuracy rates, especially after the second classification step, and is fast even on a moderately performing laptop, paving the way for a future real-time process.

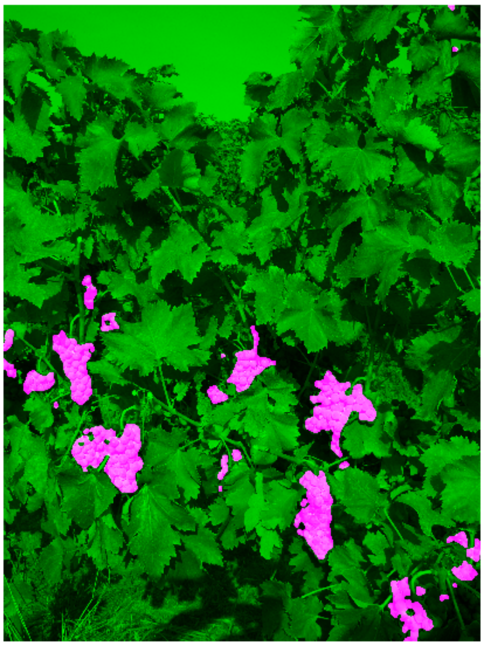

Figure 8. Blue grapes: Final classification map superposed on the original RGB image in Figure 4 (Left)—pixels classified as blue grapes are in magenta.

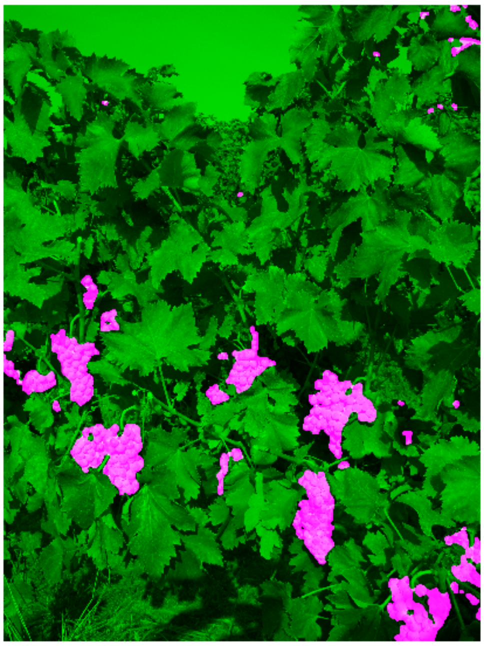

Figure 9. Blue grapes: Final classification of the image in Figure 4 (Left) and using training sets composed of 10 examples. The classification accuracy is 95.3%—pixels classified as blue grapes are in magenta.

Table 1. Blue grapes: Classification accuracy (%) for a decreasing size of the training sets used in both phases of the proposed method. The training time and computing time, measured in seconds (*s*) and required for building the classification maps, have also been provided for each step of the proposed method. The last column refers to the processing time of the whole procedure.

	N° POINTS	**ACCURACY**		**TIME (*s*)**	
			training	map	total
First classification	100	93.6	1.296	39.602	
Second classification	100	96.5	3.064	23.93	67.89
First classification	90	93.0	1.037	33.742	
Second classification	90	96.6	2.523	17.588	54.89
First classification	80	93.5	1.088	30.720	
Second classification	80	96.6	2.065	23.758	57.63
First classification	70	93.4	1.105	30.824	
Second classification	70	96.3	2.052	15.705	49.67
First classification	60	93.4	1.050	31.264	
Second classification	60	96.3	2.057	17.725	52.09
First classification	50	93.5	1.056	31.078	
Second classification	50	96.2	2.158	15.872	50.16
First classification	40	93.7	1.031	30.749	
Second classification	40	96.2	2.061	14.055	47.89
First classification	30	90.8	1.060	30.626	
Second classification	30	95.9	2.301	17.464	51.45
First classification	20	94.3	1.099	30.617	
Second classification	20	95.2	2.089	13.708	47.51

3.2. *Yellow Grapes*

With regard to the yellow grapes, again, the steps in Figure 3 were performed. In particular, the input image was converted into the complementary CMY and filtered with the same filter. The feature vector, whose components are defined in Equations (8) and (7) and shown in Figure 10, was employed in the first classification whose result is depicted in Figure 11. The second classification in the filtered HSV space led to the M_2 map in Figure 12 (Left), while Figure 12 (Right) shows its post processed version. The final result is shown in Figure 13.

Table 2. Blue grapes: Classification accuracy (%) for training sets composed of only 10 samples in both classifications—the result of six different runs is presented; each run requires about 51 s.

	N° POINTS	**ACCURACY (%)**
First classification	10	91.6
Second classification	10	96.0
First classification	10	92.5
Second classification	10	95.7
First classification	10	85.8
Second classification	10	95.7
First classification	10	95.2
Second classification	10	95.3
First classification	10	93.8
Second classification	10	95.6
First classification	10	92.1
Second classification	10	96.5

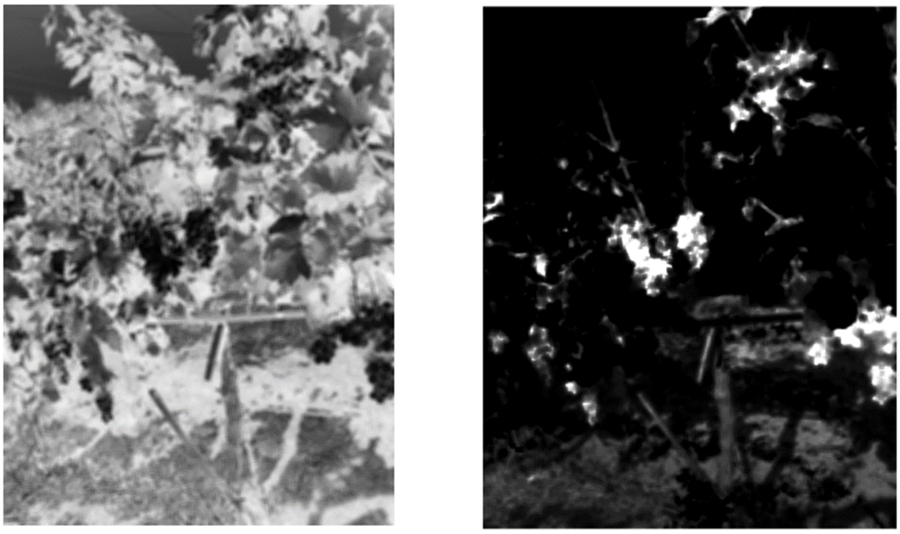

Figure 10. Yellow grapes: (**Left**) luminance L_y as defined in Equation (8) and (**Right**) contrast Con_y as defined in Equation (7).

Figure 11. Yellow grapes: M_1 map relative to the first classification superposed on the original RGB image—pixels classified as yellow grapes are in pink.

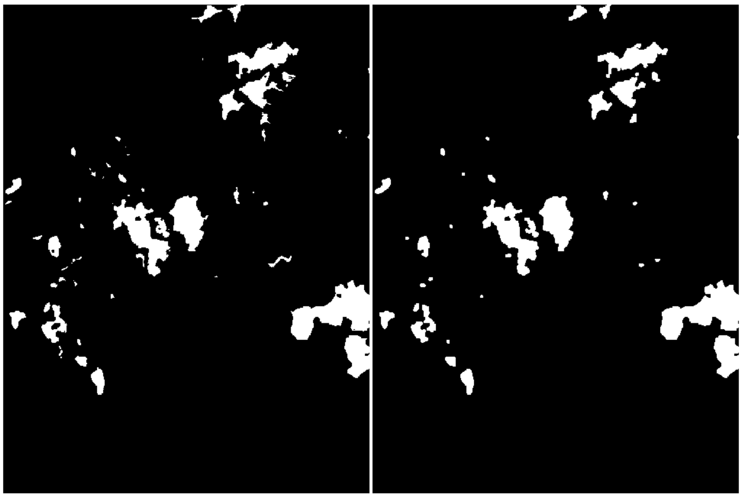

Figure 12. Yellow grapes: (**Left**) M_2 map and (**Right**) its post-processed version.

Figure 13. Yellow grapes: Final classification map superposed on the original RGB image in Figure 4 (Right)—pixels classified as yellow grapes are in pink.

As for blue grapes, many classifications were made, and only a subset of them are shown in this section. In Table 3, the classification accuracy for a decreasing size of the training set is reported. The need for a small training set is confirmed even in this case. However, Table 4 shows that having only 10 points in the training set is not always sufficient to guarantee a classification accuracy greater than 95%. This can be easily explained with the intuitive observation that a fast recognition of yellow grapes on a general yellowish/greenish background makes the problem more difficult than for blue grapes.

A larger training set is required in this case. Table 5 refers to some trials where 20 points were used for training. As can be observed, 20 points in the training set guarantees a final classification rate greater than 95% in this case. Finally, Figure 14 gives evidence of the better quality of the classification provided by the proposed method using 20 points in the training set when compared to the one achieved using only 10 points. As far it concerns the computing time, this is comparable to that required for processing blue grapes.

Table 3. Yellow grapes: Classification accuracy (%) for a decreasing size of the training set in both phases. The training time and computing time, measured in seconds (*s*), required for building the classification maps were also provided for each step of the proposed method. The last column refers to the processing time of the whole procedure.

	N° POINTS	**ACCURACY**		**TIME** (*s*)	
			training	map	total
First classification	100	92.2	1.20	31.44	
Second classification	100	95.8	2.04	16.07	50.74
First classification	90	92.2	1.037	32.07	
Second classification	90	95.8	2.271	15.413	50.79
First classification	80	91.7	1.053	31.930	
Second classification	80	95.8	2.510	17.748	53.24
First classification	70	91.8	1.017	31.459	
Second classification	70	95.8	2.170	17.117	51.76
First classification	60	91.8	1.107	30.480	
Second classification	60	95.7	2.070	16.440	50.09
First classification	50	91.6	1.084	31.217	
Second classification	50	95.6	2.240	16.308	50.85
First classification	40	92.3	1.111	30.643	
Second classification	40	95.5	2.030	15.244	49.03
First classification	30	92.6	2.275	35.295	
Second classification	30	95.2	3,058	21,588	62.22
First classification	20	92.3	1.418	36.640	
Second classification	20	95.0	2.459	17.149	57.67

Table 4. Yellow grapes: Classification accuracy (%) for only 10 points in the training set—the results for six different runs are presented.

	N° POINTS	**ACCURACY** (%)
First classification	10	93.0
Second classification	10	94.8
First classification	10	91.1
Second classification	10	94.3
First classification	10	89.9
Second classification	10	94.5
First classification	10	88.0
Second classification	10	93.7
First classification	10	92.2
Second classification	10	95.1
First classification	10	94.5
Second classification	10	94.8

Table 5. Yellow grapes: Classification accuracy (%) for 20 points in the training set.

	N° POINTS	ACCURACY (%)
First classification	20	93.5
Second classification	20	95.3
First classification	20	92.5
Second classification	20	95.5
First classification	20	92.9
Second classification	20	95.8
First classification	20	91.3
Second classification	20	95.2
First classification	20	92.3
Second classification	20	95.0
First classification	20	92.0
Second classification	20	95.1

Figure 14. Yellow grapes: Final classification result for the image with 10−points (**Left**) and 20−points (**Right**) training set—pixels classified as yellow grapes are in pink.

3.3. Comparative Studies and Discussions

The proposed approach is compared with the one presented in [1] since it is the most similar in spirit to the proposed one. This approach is based on three main phases: pre-processing, dataset training, and classification through an SVM classifier. Its main ingredients are the HSV space (as in the proposed approach) and Otsu's threshold along with some morphological operations on the resulting binary maps. Successively, regions of interest (ROIs) are found and a classification on vectors involving various (i.e., 14) features, both geometrical and statistical, is performed.

Specifically, the adopted features are: closeness, extent, compactness, texture, H mean, H and S average contrast, H S and V smoothness, S third moment, H and V uniformity and H and S entropy. It is worth outlining that these features were selected among a larger set via two well-known techniques: ReliefF algorithm [97] and sequential feature selection [34]. The selected features are used for the SVM classification. The accuracy rates achieved on the 'in-the-wild' images in Figure 15 by the method in [1] were, respectively, 52% and 50%.

As can be observed, even with the use of a more populated training set (120 for the first classification and 179 for the second one), the method in [1] achieved a lower classification accuracy compared with the proposed approach, reaching accuracy rates of about 95% for the two images in Figure 15 using 20-point training sets in both classification steps.

This is the consequence of the use of the optimized visual-perception-based features in the proposed approach.

In fact, color perception and visual contrast play a significant role in the determination of the visual saliency of the objects under study, which represents one of the main ingredients in the naked eye yield analysis performed by a winemaker. As Figures 5 (Right) and 10 (Right) show, the proposed visual contrast allows greatly emphasizing the grapes with respect to the remaining image components. The role of SVM-based classification is then to automatically define the separation threshold in a data-driven fashion. On the other hand, the ML-based approach benefits from the definition of specific and relevant features for the object under study, so the use of small training sets is allowed.

In addition to the benefits discussed in the previous subsections, the selection of points in the training set must be accurate in order to prevent misclassifications—this recommendation becomes fundamental whenever the proposed procedure is embedded in a smart application that enables the winemaker to retrain the classifier. On the other hand, this can be less troublesome than acquiring a large number of images as required in DNN-based approaches.

Finally, with regard to the computing time, although the proposed procedure shows some merits with respect to competing methods due to the very simple operations employed, some further work is required to optimized some of its steps, especially the testing phase. Region-based instead of the proposed pixelwise strategies could be employed for promoting real-time processing.

Figure 15. Blue grapes: (**Top**) Two test images. (**Bottom**) Grape classification for the method in [1].

4. Conclusions

In this paper, we proposed a cascaded classification method of grape bunches. Its main peculiarity is the use of the human perception mechanism of early vision in order to define proper feature vectors to use as input for the two classifiers. This property enables replicating the almost straightforward grape-bunch detection process that is performed by a winemaker. As a consequence, a very small training set (a small number of image pixels) can be used in the learning phase of the classification procedures. In addition, the method is robust to "in-the-wild" videos that are acquired in uncontrolled acquisition conditions.

These two ingredients make the proposed method implementable on smart devices in a user-friendly fashion, making it directly usable and updatable even by the winemaker. When compared with similar methods, the proposed approach showed that the selection of a smaller number of features and the adoption of a small training set are possible by adopting visual-perception-based features that have been 'naturally' optimized over hundreds of thousands of years. Future research will be devoted to refining the proposed computational procedure to increase its accuracy as well as to defining a unified method for both blue and yellow grapes.

Author Contributions: Conceptualization, V.B., G.D. and D.V.; methodology, V.B. and D.V.; software, G.D.; validation, G.D. and D.V.; formal analysis, V.B. and G.D.; investigation, G.D. and D.V.; resources,

D.V.; data curation, V.B., G.D. and D.V.; writing—original draft preparation, G.D. and D.V.; writing—review and editing, V.B. and D.V. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially funded by Regione Lazio and Sigma Consulting s.r.l (Industrial Ph.D. of Giulia Dominijanni, 'Intervento per il rafforzamento della ricerca nel Lazio-incentivi per i dottorati di innovazione per le imprese'—CUP B85F21000060005). This research was accomplished within RITA (Research ITalian network on Approximation) and the Italian national research group GNCS (INdAM).

Institutional Review Board Statement: Not applicable

Informed Consent Statement: Not applicable

Data Availability Statement: Data available on request.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Liu, S.; Whitty, M. Automatic grape bunch detection in vineyards with an SVM classifier. *J. Appl. Log.* **2015**, *13*, 643–653.
2. Lu, S.; Liu, X.; He, Z.; Karkee, M.; Zhang, X. Swin-Transformer-YOLOV5 For Real-Time Wine Grape Bunch Detection. *Remote Sens.* **2022**, *14*:5853.
3. Dami, I.; Sabbatini, P. *Crop Estimation of Grapes*; Tech. rep. HYG-1434-11; The Ohio State University: Columbus, OH, USA, 2011.
4. Stephen Martin, R.D.; Dunn, G. *How to Forecast Wine Grape Deliveries*; Technique report; Department of Primary Industries:State of Victoria 2003.
5. Gu, J.; Liu, T.; Wang, X.; Wang, G.; Cai, J.; Chen, T. Recent Advances in Convolutional Neural Networks. *Pattern Recognit.* **2018**, *77*, 354–377.
6. Jiao, L.; Zhang, F.; Liu, F.; Yang, S.; Li, L.; Feng., Z.; Qu, R. A survey of deep learning-based object detection. *IEEE Access* **2021**, *10*,20118-20134.
7. Huang, Z.; Zhang, P.; Liu, R.; Li, D. Immature apple detection method based on improved yolov3. *ASP Trans. Internet Things* **2021**, *1*, 9–13.
8. Chen, J.; Wang, Z.; Wu, J.; Hu, Q.; Zhao, C.; Tan, C.; Teng, L.; Luo, T. An improved yolov3 based on dual path network for cherry tomatoes detection. *J. Food Process. Eng.* **2021**, *44*. https://doi.org/10.1111/jfpe.13803.
9. Chen, Y.; Li, J.; Xiao, H.; Jin, X.; Yan, S.; Feng, J. Dual path networks. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 4470–4478 .
10. Lu, S.; Chen, W.; Zhang, X.; Karkee, M. Canopy-attention-yolov4-based immature/mature apple fruit detection on dense-foliage tree architectures for early crop load estimation. *Comput. Electron. Agric.* **2022**, *193*, 106696.
11. Woo, S.; Park, J.; Lee, J.Y.; Kweon, S. Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, September 2018.
12. Aggarwal, C.C. *Neural Networks and Deep Learning, A Textbook*; Springer: Berlin/Heidelberg, Germany, 2018.
13. Marani, R., Milella, A., Petitti, A., Reina, G. Deep learning-based image segmentation for grape bunch detection *Precis. Agric.* **2019**, *19*, 791–797.
14. Marani, R.; Milella, A.; Petitti, A., Reina, G. Deep neural networks for grape bunch segmentation in natural images from a consumer-grade camera. *Precis. Agric.* **2021**, *22*, 387–413.
15. Winkler, S. *Digital Video Quality—Vision Models and Metrics*; John Wiley & Sons, Ltd.: Hoboken, NJ, USA, 2005.
16. Solomon, R.L. The Opponent-Process Theory of Acquired Motivation: The Costs of Pleasure and the Benefits of Pain. *Am. Psychol.* **1980**, *35*, 691–712.
17. Pridmore, R. W. Single cell spectrally opposed responses: Opponent colours or complementary colours? *J. Opt.* **2012**, *42*, 8–18.
18. Mazur, J.E. *Learning and Behavior*, seventh ed.; Pearson: Boston, MA, USA, 2013.
19. Leknes, S.; Brooks, J.C.W.; Wiech, K.; Tracey, I. Pain relief as an opponent process: A psychophysical investigation. *Eur. J. Neurosci.* **2008**, *28*, 794–801.
20. Bruni, V.; Rossi, E.; Vitulano, D. Jensen-Shannon divergence for visual quality assessment. *Signal Image Video Process.* **2013**, *7*, 411–421.
21. Bruni, V.; Vitulano, D.; Wang, Z. Special issue on human vision and information theory. *Signal Image Video Process.* **2013**, *7*, 389–390.
22. Ramella, G. Evaluation of quality measures for color quantization. *Multimed. Tools Appl.* **2021**, *80*, 32975–33009.
23. Nuske, S.; Gupta, K.; Narasimhan, S.; Singh, S. Modeling and calibrating visual yield estimates in vineyards. *Field Serv. Robot.* **2014**, 343–356, doi: 10.1007/978-3-642-40686-7_23.
24. Loy, G.; Zelinsky, A. Fast radial symmetry for detecting points of interest. *IEEE Trans. Pattern Anal. Mach. Intell.* **2003**. *25*, 959–973.

25. Liu, S.; Marden, S.; Whitty, M. Towards automated yield estimation in viticulture. In Proceedings of the Australasian Conference on Robotics and Automation, Sydney, Australia, December 2013; pp. 2–4.
26. Correa, C.; Valero, C.; Barreiro, P.; Tardaguila, J.; Diago, M. A comparison of fuzzy clustering algorithms applied to feature extraction on vineyard. In *Avances en Inteligencia Artificial*; Lozano, J., Gomez, J., Moreno, J., Eds.; Springer 2011; Volume 1, pp. 1–10.
27. Dey, D.; Mummert, L.; Sukthankar, R. Classification of plant structures from uncalibrated image sequences. In Proceedings of the 2012 IEEE Workshop on Applications of Computer Vision, WACV, Breckenridge, CO, USA, 9–11 January 2012; pp. 329–336.
28. Farias, C.C.; Ubierna, C.V.; Elorza, P.B. Characterization of vineyard's canopy through fuzzy clustering and SVM over color images. In Proceedings of the International Conference of Agricultural Engineering, Valencia, Spain, July 2012.
29. Diago, M.; Correa, C.; Millán, B.; Barreiro, P. Grapevine yield and leaf area estimation using supervised classification methodology on RGB images taken under field conditions. *Sensors* **2012**. *12*, 16988–17006.
30. Tardaguila, J.; Diago, M.; Millán, B. Applications of computer vision techniques in viticulture to assess canopy features, cluster morphology and berry size. In *International Workshop on Vineyard Mechanization and Grape and Wine Quality*; March 2013; Volume 978.
31. Chamelat, R.; Rosso, E.; Choksuriwong, A.; Rosenberger, C.; Laurent, H.; Bro, P. Grape detection by image processing. In Proceedings of the IECON 2006—32nd Annual Conference on IEEE Industrial Electronics, Paris, France, 6–10 November 2006; pp. 3697–3702.
32. Reis, M.J.C.S.; Morais, R.; Peres, E.; Pereira, C.; Contente, O.; Soares, S.; Valente, A.; Baptista, J.; Ferreira, P.J.S.G.; Bulas Cruz, J. Automatic detection of bunches of grapes in natural environment from color images. *J. Appl. Log.* **2012**, *10*, 285–290.
33. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**. *37*, 1904–1916.
34. Kittler, J. Feature set search algorithms. *Pattern Recognit. and Signal Process.* **1978**, 41–60.
35. Cristianini, N.; Shawe-Taylor, J. *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*; Cambridge University Press: Cambridge, UK, 2000.
36. Zou, Z.; Shi, Z.; Guo, Y.; Ye, J. Object detection in 20 years: A survey. *arXiv Prepr.* **2019**, arXiv:1905.05055.
37. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, October 2016.
38. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *42*, 2980–2988.
39. Tian, Z.; Shen, C.; Chen, H.; He, T. Fcos: Fully convolutional one-stage object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, October 2019; pp. 9627–9636.
40. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, August, 2020; pp. 213–229.
41. Tan, M.; Pang, R.; Le, Q.V. Efficientdet: Scalable and efficient object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, June 2020; pp. 10781–10790.
42. Bochkovskiy, A.; Wang, C.Y.; Liao, H. Yolov4: Optimal speed and accuracy of object detection. *arXiv Prepr.* **2020**, arXiv:2004.10934.
43. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv Prepr.* **2018**, arXiv:1804.02767.
44. Jocher, G.; Chaurasia, A.; Stoken, A.; Borovec, J.; Kwon, Y.; Michael, K.; Fang, J.; Wong, C.; Montes, D.; Wang, Z.; et al. ultralytics/yolov5: v6.2 - YOLOv5 Classification Models, Apple M1, Reproducibility, ClearML and Deci.ai integrations. 2022, doi:10.5281/zenodo.7002879
45. Wang, C.Y.; Bochkovskiy, A.; Liao, H. Scaled-yolov4: Scaling cross stage partial network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, **2021**; pp. 13029–13038.
46. Sozzi, M.; Cantalamessa, S.; Cogato, A.; Kayad, A.; Marinello, F. Automatic Bunch Detection in White Grape Varieties Using YOLOv3, YOLOv4, and YOLOv5 Deep Learning Algorithms. *Agronomy* **2022**, *12*, 319.
47. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, , Columbus, OH, USA, June 2014; pp. 580–587.
48. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Boston, June 2015; pp. 1440–1448.
49. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 91–99.
50. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, Hawaii, June 2017; pp. 2117–2125.
51. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. Centernet: Keypoint triplets for object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea (South), 2019; pp. 6569–6578.
52. Gao, F.; Fu, L.; Zhang, X.; Majeed, Y.; Li, R.; Karkee, M.; Zhang, Q. Multi-class fruit-on-plant detection for apple in snap system using faster r-cnn. *Comput. Electron. Agric.* **2020**, *176*, doi: 10.1016/j.compag.2020.105634.
53. Tu, S.; Pang, J.; Liu, H.; Zhuang, N.; Chen, Y.; Zheng, C.; Wan, H.; Xue, Y. Passion fruit detection and counting based on multiple scale faster r-cnn using rgb-d images. *Precis. Agric.* **2020**. *21*, 1072–1091.

54. Fu, L.; Gao, F.; Wu, J.; Li, R.; Karkee, M.; Zhang, Q. Application of consumer rgb-d cameras for fruit detection and localization in field: A critical review. *Comput. Electron. Agric.* **2020**, *177*, doi: 10.1016/j.compag.2020.105687.
55. Koirala, A.; Walsh,K.B.; Wang, Z.; McCarthy, C. Deep learning for real-time fruit detection and orchard fruit load estimation: Benchmarking of 'mangoyolo'. *Precis. Agric.* **2019**, *20*, 1107–1135.
56. Tian, Y.; Yang, G.; Wang, Z.; Wang, H.; Li, E.; Liang, Z. Apple detection during different growth stages in orchards using the improved yolo-v3 model. *Comput. Electron. Agric.* **2019**, *157*, 417–426.
57. Khan, S.; Naseer, M.; Hayat, M.; Zamir, S.W.; Khan, F.S.; Shah, M. Transformers in vision: A survey. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–41.
58. Li, Y.; Mao, H.; Girshick, R.; He, K. Exploring plain vision transformer backbones for object detection. *arXiv Prepr.* **2022**, arXiv:2203.16527.
59. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021; pp. 10012–10022.
60. Liu, Z.; Tan, Y.; He, Q.; Xiao, Y. Swinnet: Swin transformer drives edge-aware rgb-d and rgb-t salient object detection. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *32*, 4486–4497.
61. Hatamizadeh, A.; Nath, V.; Tang, Y.; Yang, S.; Roth, H.R.; Xu, D. Swin unetr: Swin transformers for semantic segmentation of brain tumors in mri images. In *International MICCAI Brainlesion Workshop*; Springer International Publishing: Cham, Switzerland, 2022; pp. 272–284.
62. Jannat, F.E.; Willis, A.R. Improving classification of remotely sensed images with the swin transformer. In Proceedings of the SoutheastCon, Mobile, AL, USA, 26 March–3 April 2022; pp. 611–618.
63. Zhao, H.; Jia, J.; Koltun, V. Exploring self-attention for image recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020; pp. 10076–10085.
64. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 740–755.
65. Naseer, M.M.; Ranasinghe, K.; Khan, S.H.; Hayat, M.; Khan, F.S.; Yang, M.H. Intriguing properties of vision transformers. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 23296–23308.
66. Zheng, H.; Wang, G.; Li, X. Swin-mlp: A strawberry appearance quality identification method by swin transformer and multi-layer perceptron. *J. Food Meas. Charact.* **2022**, *16*, 1–12.
67. Wang, F.; Rao, Y.; Luo, Q.; Jin, X.; Jiang, Z.; Zhang, W.; Li, S. Practical cucumber leaf disease recognition using improved swin transformer and small sample size. *Comput. Electron. Agric.* **2022**, *199*, 107163.
68. Wang, J.; Zhang, Z.; Luo, L.; Zhu, W.; Chen, J.; Wang, W. Swingd: A robust grape bunch detection model based on swin transformer in complex vineyard environment. *Horticulturae* **2021**, *7*, 492.
69. Sozzi, M.; Cantalamessa, S.; Cogato, A.; Kayad, A.; Marinello, F. wGrapeUNIPD-DL: An open dataset for white grape bunch detection. *Data Brief.* **2022**, *43*, 108466.
70. Sentís, M.A.; Vélez, S.; Valente, J. Dataset on UAV RGB videos acquired over a vineyard including bunch labels for object detection and tracking. *Data Brief.* **2022**, *46*, 108848.
71. Mallat, S. *A Wavelet Tour of Signal Processing*, second ed.; Elsevier: Amsterdam, The Netherlands, 1999.
72. Burt, P.I.; Adelson, E.H. The Laplacian pyramid as a compact image code. *IEEE Trans. Commun.* **1983**, *31*, 532–540.
73. Mallat, S. Multiresolution approximations and wavelet orthonormal bases of L2(R). *Trans. Amex Math. Soc.* **1989**, *315*, 69–87.
74. Meyer, Y. *Wavelets and Operators. Advanced Mathematics*; Cambridge University Press: Cambridge, UK, 1992.
75. Mante, V.; Frazor, R.A.; Bonin, V.; Geisler, W.S.; Carandini, M. Independence of luminance and contrast in natural scenes and in the early visual system. *Nat. Neurosci.* **2005**, *8*, 1690–1697.
76. Bruni, V.; Rossi, E.; Vitulano, D. On the equivalence between jensen-shannon divergence and michelson contrast. *IEEE Trans. Inf. Theory* **2012**, *58*, 4278–4288.
77. Bruni, V.; Crawford, A.; Vitulano, D. Visibility based detection of complicated objects: A case study. In Proceedings of the third European Conference on Visual Media Production, CVMP 2006, London, UK, 29–30 November 2006; pp. 55–64.
78. Bruni, V.; Crawford, A.; Kokaram, A.; Vitulano, D. Semi-transparent blotches removal from sepia images exploiting visibility laws. *Signal Image Video Process.* **2013**, *7*, 11–26.
79. Bernstein, D.A. *Essentials of Psychology*, fourth ed.; Cengage Learning: Boston, MA, USA, 2011.
80. Zeki, S.; Cheadle, S.; Pepper, J.; Mylonas, D. The constancy of colored after-images. *Front. Hum. Neurosci.* **2017**, *11*, 229.
81. Shapley, R.M.; Enroth-Cugell, C. Visual adaptation and retinal gain controls. *Prog. Retin. Res.* **1984**, *3*, 263–346.
82. Troy, J.B.; Enroth-Cugell, C.X.; Y ganglion cells inform the cat's brain about contrast in the retinal image. *Exp. Brain Res,* **1993**, *93*, 383–390.
83. Rodieck, R.W. *The First Steps in Seeing*; Sinauer: Sunderland, MA, USA, 1998.
84. Shapley, R.M., Victor, J.D. The effect of contrast on the transfer properties of cat retinal ganglion cells. *J. Physiol.* **1978**, *285*, 275–298.
85. Victor, J. The dynamics of the cat retinal X cell centre. *J. Physiol.* **1997**, *386*, 219–246.
86. Baccus, S.A., Meister, M. Fast and slow contrast adaptation in retinal circuitry. *Neuron* **2002**, *36*, 909–919.
87. Demb, J.B. Multiple mechanisms for contrast adaptation in the retina. *Neuron* **2002**, *36*, 781–783.

88. Kaplan, E.; Purpura, K.; Shapley, R. Contrast affects the transmission of visual information through the mammalian lateral geniculate nucleus. *J. Physiol.* **1987**, *391*, 267–288.
89. Sclar, G.; Maunsell, J.H.; Lennie, P. Coding of image contrast in central visual pathways of the macaque monkey. *Vision Res.* **1990**, *30*, 1–10.
90. Bonin, V.; Mante, V.; Carandini, M. The suppressive field of neurons in lateral geniculate. *J. Neurosci.* **2005**, *25*, 10844–10856.
91. Graf, H.; Cosatto, E.; Bottou, L.; Dourdanovic, I.; Vapnik, V. Parallel support vector machines: The cascade svm. *Adv. Neural Inf. Process. Syst.* **2004**, *17*, 521–528.
92. Ze-Shen, L.; Zhi-Song, P. Research on parallel svm algorithm based on spark. *Comput. Sci.* **2016**, *43*, 238–242.
93. Available online: https://en.wikipedia.org/wiki/HSL_and_HSV (accessed on 21 December 2022).
94. Available online: https://webaim.org/resources/contrastchecker/ (accessed on 21 December 2022).
95. Bruni, V.; Ferrara, P.; Vitulano, D. Removal of Color Scratches from Old Motion Picture Films Exploiting Human Perception. *EURASIP J. Adv. Signal Process.* **2008**, *2008*, 1–9.
96. Gonzalez, R.C. *Digital Image Processing*, fourth ed.;Woods, R.E., Ed.; Pearson: New York, NY, USA, 2018; pp. 138–140.
97. Robnik-Šikonja, M.; Kononenko, I. Theoretical and empirical analysis of ReliefF and RReliefF. *Mach. Learn.* **2003**, *53*, 23–69.