Journal of Universal Computer Science, vol. 28, no. 3 (2022), 311-328
submitted: 4/11/2021, accepted: 11/1/2022, appeared: 28/3/2022 CC BY-ND 4.0

Water stress classification using Convolutional Deep Neural Networks

Lerina Aversano
(University of Sannio, Benevento, Italy
https://orcid.org/0000-0003-2436-6835, aversano@unisannio.it)

Mario Luca Bernardi
(University of Sannio, Benevento, Italy
https://orcid.org/0000-0002-3223-7032, bernardi@unisannio.it)

Marta Cimitile
(UnitelmaSapienza University, Rome, Italy
https://orcid.org/0000-0003-2403-8313, marta.cimitile@unitelmasapienza.it)

Abstract: In agriculture, given the global water scarcity, optimizing the irrigation system have become a key requisite of any semi-automatic irrigation scheduling system. Using efficient assessment methods for crop water stress allows reduced water consumption as well as improved quality and quantity of the production. The adoption of Neural Network can support the automatic in situ continuous monitoring and irrigation through the real-time classification of the plant water stress. This study proposes an end-to-end automatic irrigation system based on the adoption of Deep Neural Networks for the multinomial classification of tomato plants' water stress based on thermal and optical aerial images. This paper proposes a novel approach that cover three important aspects: (*i*) joint usage of optical and thermal camera, captured by un-manned aerial vehicles (UAVs); (*ii*) strategies of image segmentation in both thermal imaging used to obtain images that can remove noise and parts not useful for classifying water stress; (*iii*) the adoption of deep pre-trained neural ensembles to perform effective classification of field water stress. Firstly, we used a multi-channel approach based on both thermal and optical images gathered by a drone to obtain a more robust and broad image extraction. Moreover, looking at the image processing, a segmentation and background removal step is performed to improve the image quality. Then, the proposed VGG-based architecture is designed as a combination of two different VGG instances (one for each channel). To validate the proposed approach a large real dataset is built. It is composed of 6000 images covering all the lifecycle of the tomato crops captured with a drone thermal and optical photocamera. Specifically, our approach, looking mainly at leafs and fruits status and patterns, is designed to be applied after the plants has been transplanted and have reached, at least, early growth stage (covering vegetative, flowering, friut-formation and mature fruiting stages).

Keywords: intelligent systems, deep learning, smart irrigation, precision agriculture, digital agriculture
Categories: I.2.10, I.2.8, I.4.1
DOI: 10.3897/jucs.80733

1 Introduction

The consumption of worldwide water largely derives from agriculture [Miras-Avalos et al., 2019] and is highly dependent on the capability to compute crop water requirements [Al-Ghobari and Mohammad, 2011]. Optimal agriculture's water management is, indeed, a critical aspect while it allows for sustainable advancement in agriculture by ensuring the right quality and quantity of crops.

Physiological parameters of tomato plants include stem and leaf water potential, and stomatal conductance. These have been used as indicators to evaluate crop water stress for long time. Specifically, irrigation plan is defined on the direct measurement of these physiological indicators (e.g., evapotranspiration-based estimation, soil moisture and plant water potential measurement). However these traditional approaches are highly demanding and make use of instruments that needs careful installation and maintenance. From this point of view, non-invasive real-time crop water stress monitoring is enormously needed to support irrigation management reducing the impact on field management. According to this, several approaches are developed in the last years to support farmers to identify the right irrigation at the right place and at the right time [Gallardo et al., 2020]. These approaches are mainly based on the analysis of a set of hand-crafted features designed by experts to extract relevant information in a not automatic manner [Rinaldi and He, 2014]. More recently, the use of image processing combined with machine learning for the classification of water stress of crops is explored [Chandel et al., 2020, Zhuang et al., 2017]. Differently from the visual scoring-based methods, optical imaging-based approaches allow to detect changes (i.e., water stress) in the plant physiology rapidly and without any contact [Gao et al., 2020]. Mainly, image acquisition is performed by using fixed camera systems that allow automatic in situ continuous monitoring. However, the gathered images describe a single perspective of the field depending on the angle of the cameras.

Thermal images have been used [Hoffmann et al., 2016, Lima et al., 2016, Petrie et al., 2019] to calculate water stress indices such as CWSI for different types of crops at different scales, aiding various image processing approaches for segmenting green vegetation in thermal imaging. This study proposes a different approach to build an end-to-end intelligent irrigation system based on the adoption of Deep Neural Networks for the multinomial dicrete classification (*water excess*, *water deficit*, and *well-watered*) of tomato plants water stress from a large dataset of aerial images. The choice of tomato plants for our application scenario is supported by the awareness that in tomato crops, the irrigation process is very critical while the water stress conditions highly influence the tomato quality [Feng Wang et al., 2011]. The images have been collected with predefined scheduling to cover all the plant lifecycle. This scheduling allowed us to consider the water status depending on the main growth phase of the plant. Of course, an agronomist actively supported the construction of the dataset. Overall, several novelties are introduced by this study. Firstly, we used a multi-channel approach based on both thermal and optical images extracted by a drone system to obtain a more robust and broad image extraction. According to this, the proposed VGG-based architecture is designed as a combination of two different VGG instances (one for each channel). Moreover, looking at the image processing, a segmentation and background removal step is performed to

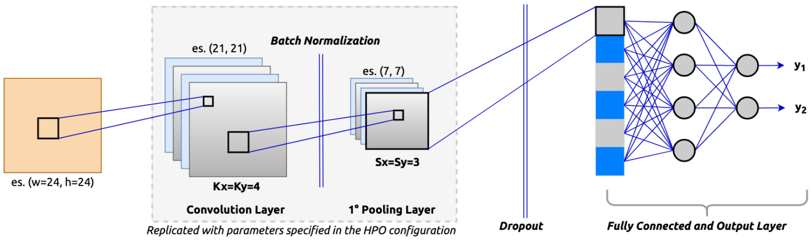


Figure 1: Example of Convolutional Neural Network.

improve the image quality. Finally, to validate the proposed approach a large real dataset is built. It is composed of 6000 images covering all the lifecycle of the tomato crops captured with a drone system.

In the rest of the document the following sections are reported: Section 2 discusses some background notions; Section 3 provides an overview and a comparison with the related work; Section 4 describe the overall approach; Section 5 describes the experiment setting and the obtained results. Finally, in Section 6 the conclusions are reported.

2 Background

Deep Learning neural networks take inspiration from the way biological nervous systems process information (artificial perceptrons can be seen as human neurons). These networks model the studied problem by means of several functions that perform a hierarchical representation of the data based on several abstraction levels composed of various artificial perceptrons [Deng et al., 2014]. In particular, deep neural networks consist of several hidden layers: in each step, the input data are transformed with a slightly more abstract and composite representation. The layers can be seen as hierarchies of concepts, that can be used to perform pattern classification and feature learning. Similarly to the training of a traditional neural network, for a DL network, the forward and backward phases are concerned. In the forward step, the nodes' activation signals flow from the input to the output layer. In the backward phase, weights and biases are possibly modified to improve the network overall performance.

Compared with Machine Learning (ML), DL is more suitable in complex domains while it allows one to gradually improve the overall performance by adopting blackbox models. As a matter of fact, DL algorithms are very widespread in the fields of computer vision, natural language processing, speech recognition, health informatics, audio recognition, social network filtering, and machine translation.

2.1 The Convolutional Neural Network

A Convolutional Neural Network (CNN) is a type of feed-forward neural network, consisting of several stages, each characterized by the specialization of different functionalities [Ahmad et al., 2019]. In particular, unlike a normal neural network, the layers of a CNN have neurons arranged in three dimensions: width, height, depth, whereby depth means the third dimension of an activation volume. Besides, neurons in one layer are connected only to a small region of the layer before it, rather than to all neurons in a fully connected way. Figure 1 shows the operation diagram of a CNN, made up of a chain of ordered blocks, wherein each block identifies one level of the network that transforms one volume of activations into another one through a differentiable function:

- **Input layer**: represents the set of data in the form of numbers to be analyzed;
- **Convolutional layer**: extracts the main features through the use of filters. It is composed of several convolutional kernels that allow to compute the feature maps. In particular, each neuron of the feature map is connected to a neuron of a region of input layer trhought convolution. Looking to the Figure 1, as example, the input layer is a matrix of 24x24 and generates a feature map at the Convolutional layer having size of 21x21 since the kernel coordinates equal to four (kx=ky=4). More formally, the feature maps are obtained using different kernels that is composed of spatial locations of the input. According to this the feature value is:

  $$\mathbf{z^l_{i,j,k}} = {w^l_k}^T X^l_{i,j} + b^l_k \quad (1)$$

  where: w^l_k and b^l_k are repectively the weight vector and the bias term, and $X^l_{i,j}$ is the input patch at location (i,j) in the k-th feature map of l-th layer.
- **Pooling layer**: allows the identification of the presence of the study characteristic in the previous level. It simplifies and roughens the data, keeping the characteristic used by the convolutional level, and performs an aggregation of information, generating smaller feature maps. This level drastically reduces the spatial dimension (the height and width change but not the depth) of the input volume and the computational requirements for future levels. As shown in the figure, each feature map of a pooling layer is related to a corresponding region of the previous Convolutional layer (since the region is 3x3, from the 21x21 matrix we derive a 7x7 matrix). According to this, for each feature map, we have:

  $$\mathbf{y^l_{i,j,k}} = \text{pool}(a^l_{m,n,k}), \forall (m,n) \in R_{i,j} \quad (2)$$

  where $R_{i,j}$ is the local neighborhood around position (i,j).
- **Fully connected (FC) layer**: connects all the neurons of the previous level in order to establish the various identification classes displayed in the previous levels according to a certain probability, then performs the classification. This level basically takes an input volume (whatever the output of the convolutional level or of the ReLU or of the pool level that precedes it) and generates a vector of dimension N, i.e., the number of classes from which the program must choose.
- **Output layer**: it is the last layer of the CNN. Considering θ as the set of all the parameters of the CNN, the optimum parameters for a given task is obtained by

minimizing a loss function associated to the task. The loss of a CNN is computed as:

$$\textbf{loss} = \frac{1}{N}\sum_{n=1}^{N} l(\theta; y^{(n)}, o^{(n)}) \tag{3}$$

where N is the suitable number of input-output relations, $x(n)$ is the n-th input data, $y(n)$ is the target label corresponding to the n^{th} input data and $o(n)$ is the output of the CNN.

The activation function introduces a non-linearity into a system that is essentially calculating linear operations during convolutional operations, through the scalar product between the filter and the receptive field.

The activation value of the $z^{l}_{i,j,k}$ is computed as:

$$\mathbf{a^{l}_{i,j,k}} = a(z^{l}_{i,j,k}) \tag{4}$$

Activations functions $a(\bullet)$ are implemented by means of a ReLU (Rectified Linear Units) activation function: this step cancels all negative values, increasing the non-linear properties of the model and the global network without affecting the receptive fields of the convolutional units.

The ReLU step applies the function f (x) = max (0, x) to all values in the input volume.

In this study, the VGG19 and Xception pre-trained convolutional neural networks [Ardimento et al., 2021] have been used, indeed, the refinement of a pre-trained network with transfer learning is generally much faster and easier than training from scratch. It requires a minimal amount of data and computing resources. Transfer learning uses knowledge of one type of problem to solve similar problems. It allows one to use a pre-trained network and use it to learn a new activity. One of the advantages of a pre-trained network is that the network has already trained on a large set of features that can be usefully leveraged for new classification tasks. These features can be applied to a variety of other similar activities. For example, you can take a network trained on millions of images and re-train it to classify new objects using only hundreds of images. The most common process to implement transfer-learning using a neural network model is based upon the following step:

- acquire layers from a previously trained model;
- freeze them, to avoid overwriting weights during subsequent training steps;
- append additional trainable layers on top of the trained frozen layers that will translate the pre-trained features into a prediction for the new classification task;

Finally, the resulting model can be trained on the new dataset. After the model is trained, a fine-tuning step has to be performed to make the entire model (or only some internal layers) trainable again. This allows one to perform re-training on the new data. In the fine-tuning step, we adopted a very low learning rate to avoid overfitting and prevent too high changes to the pre-trained weights that can potentially corrupt the old features extraction. This fine-tuning step could lead to overall model improvements, by adapting the pre-trained features to better suit the new task. VGG19 was conceived and created by VGG (Visual Geometry Group, University of Oxford) for ILSVRC-2014 (ImageNet Large Scale Visual Recognition Challenge) [Simonyan and Zisserman, 2014]. It receives a fixed size RGB image (224x224 pixels) as input. The image is passed through a stack of convolutional layers using 3×3 filters, with a step size of one pixel, which covers the whole notion of an image. Spatial fill was used to preserve the spatial resolution of the image. It has 19 layers of convolution followed by three fully connected layers. The first two have 4096 channels each, the third contains 1000 channels. The final layer is a soft-max feature. All hidden layers are equipped with the rectification non-linearity, ReLU function.

3 Related work

3.1 Automatic irrigation systems

Manual irrigation is a high time-consuming activity causing consistent water dilapidation. This has fostered increasing numer of studies for the implementation of IoT automated irrigation systems. For example, authors in [Marcu et al., 2019] propose a possible solution for an IoT-based system to monitor the parameters having a direct impact on crops. Authors, in [Saraf and Gawali, 2017] describe a model including IoT for the extraction of real-time input data used for the smart irrigation monitoring and controlling. An IoT-based automated water system is also proposed in [Rao and Sridhar, 2018]. Two sensors are here used to get to the base station, data about the humidity and the temperature of the soil, the humidity, the temperature, and the duration of sunshine per day. The data storage is performed by using a cloud computing system. The above-discussed systems differ from our proposed approach because they do not include a water stress classification based on artificial intelligence or other alternative techniques. Differently, authors in [Ferrández-Pastor et al., 2018] propose a solution that includes artificial intelligence to optimize the irrigation system resources (i.e., water, energy). However, in this study, the focus is on the communication architecture more than on the data analysis. Finally, authors in [Torres-Sanchez et al., 2020] propose a solution based on the IoT allowing easier and efficient crop irrigation. The solution includes a distributed wireless sensor network covering all the regions of the farm. The extracted data are then transmitted to a common server. The approach uses machine learning algorithms to perform predictions for irrigation patterns based on crops and weather scenarios. Finally, a smart system to predict the irrigation requirements of a field using sensors to capture ground conditions parameters (i.e., soil moisture, soil temperature) and data from the Internet to forecast environmental conditions (i.e., precipitation, air temperature, humidity, and UV for the near future) is proposed in [Vij et al., 2020]. Contrarily from this study, in the proposed systems data captured by a drone are used to identify water stress conditions, and successively this data is used to evaluate identify the water effort on the base of meteo station data.

3.2 Deep learning in agricolture

Several studies propose the adoption of machine learning and deep learning techniques to evaluate the state of the health of the field. For example, authors in [Singh et al., 2018] discuss the principle of deep learning in the application for crop stress diagnosis based on images. Referring to the tomato disease, authors in [Aversano et al., 2020] use three convolutional neural networks previously trained on a similar problem to identify the type of plant disease (if any) from images of tomato leafs. Similar approaches are also proposed in [Kawasaki et al., 2015, Van Hulle et al., 2016, Brahimi et al., 2017, Agarwal et al., 2020]. These studies show the applicability of machine learning and deep learning techniques for the analysis of agricultural images. However, they differ from our study, for their goals while they are focused to provide early diagnosis and good identification of tomato diseases. Similarly, in [Tseng et al., 2018] a simulator is used to generate large datasets of synthetic aerial images of a vineyard and to study the potential for machine learning to learn local soil moisture conditions. This study differs from our proposal also for the adoption of a simulated dataset. However, several existing studies are based on the adoption of simulators [Nendel, 2014, van Diepen et al., 1989] to generate synthetic datasets driving empirical investigations but in this case, the availability of the approach in a real scenario is not ensured. Other studies are mainly aimed to perform in-field countings using deep learning techniques on aerial images. For example, in [Xu et al., 2018] a convolutional neural network (CNN) is used to detect and count cotton flowers, or blooms. The dataset is composed of aerial images clustered on the basis of their similarity. In [Lu et al., 2017] in-field counting problem of maize tassels is faced using a deep learning-based approach on a dataset of 361 field images collected in four experimental fields.

Finally, the adoption of machine learning and deep learning techniques to evaluate plant water stress by the analysis of plant images is faced in the more recent studies [Gao et al., 2020]. Authors in [Chandel et al., 2020] propose a comparative assessment of three DL models for identification of water stress (stress, no stress) in maize, okra, and soybean crops through image classification. The validation is performed on a dataset of 1200 images acquired by the camera and an accuracy of 98.3 % is obtained for the best case. Similarly, in [Zhuang et al., 2017] authors propose a model to detect water stress of maize. They use basic MLP artificial networks to assess water stress levels (well-watered, reduced watered, drought-stressed) from a set of candidate features extracted from the segmented images. Also in this case the images are captured using a camera. The accuracy of water stress recognition reached 90.39%. Even if the obtained results are encouraging in the described studies, the image acquisition is mainly performed by using camera systems that allow the in situ continuous monitoring from a fixed perspective given by the angle of the adopted camera. Differently, we adopted an automatic UAV-based system that is able to autonomously perform continuous monitoring of the field to obtain a more robust and broad image extraction. Another distinguishing aspect of our proposed approach with respect to the related work consists in the adoption of a multi-channel payload able to acquire both thermal and optical images of the same field portions. This ensures a more robust and effective classification. Specifically, we propose a VGG-based ensemble architecture designed as a combination of two different VGG instances (one for each channel).

4 Proposed approach

This section describes the proposed irrigation system and the image classification process. The proposed system uses the data captured by a drone to identify the crop condition. This data combined with data extracted by a meteo station placed in the field allows identifying the right quantity of water required by the field. In the following subsections, we describe the proposed irrigation systems, the image acquisition, and processing steps, and the image classification approach.

4.1 The proposed irrigation system

An overview of the proposed irrigation system is reported in Figure 2.

The main components are the Field Gateway and the AI Server that are connected by ethernet/wifi. The AI server contains the system platform (orange box). The platform is used by the drone operator and the field administrator that can access the dashboard respectively to insert the drone's captured images and to obtain information about the field state.

In the backend system, there are three components: the Field Server, the Flight data manager, and the Classifier. The Flight Data Manager manages the flight data as file systems. The classifier component takes the flight data and performs the classification (this process will be described in the following section). The Field Server data is then collected in a database that is connected to the backend system.

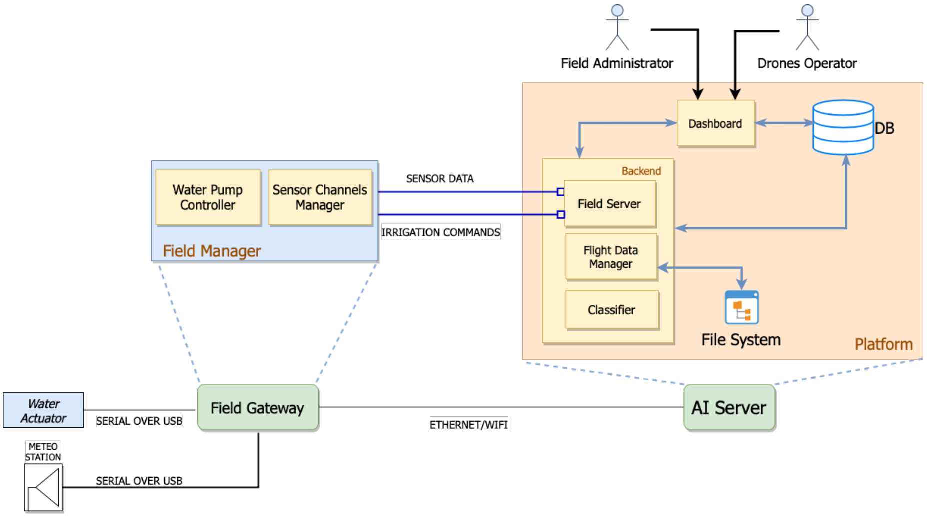


Figure 2: The automatic irrigation system.

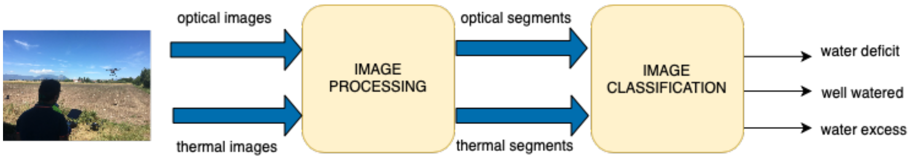


Figure 3: Image processing and classification.

The Field Server is connected to the Field Gateway and changes with it all the data extracted by the field (sensor data) and irrigation commands. However, the Field Gateway includes the water pump controller and the sensors channel manager (Field Manager).

4.2 Image acquisition and processing using spectral clustering

Figure 3 shows that the optical and thermal images acquired by a drone system are processed and segmented. The obtained optical and thermal segments are then sent to the classifier. The image acquisition process is performed through a UAV system flying over the monitored field to capture optical and thermal images of the plants.

The collected data are processed through a computer vision algorithm [Addabbo et al., 2018] running on-board the vehicle within a geo-software module, able to tag images using fixed centimeter-level positions. The Global Navigation Satellite System (GNSS) signals provide an accurate positioning enabling the automation of the entire process and allowing to geo-referencing the inspected plants by the cameras on-board the UAV. Additionally, the field is instrumented with ground markers to check and calibrate the UAV trajectory.

Finally, the collected data is sent to the remote service center for image processing.

In this phase, the thermal and optical images collected by the UAV system are processed and segmented. The collected images are firstly cleaned: all the images with low quality are cut off. Successively, the segmentation is performed. In this step, since the images contained more objects (i.e., stones, infesting grass and several plants), we segmented out plants from their backgrounds. To this aim, the spectral clustering technique [Yu and Shi, 2003] is used. This technique is used in this study because it is robust to random initialization and converges faster than the alternative clustering methods. Since each optical/termical sample contains several plants, each plant segment identified in the clustering step is generated as a separate image. Notice that, for each plant, the timestamp and the geographical position is recorded to keep track of which part of the field has been classified in a certain state.

Figure 4 reports an example of an image collected with an optical camera (left side), the image when the backgrounds are deleted and the images segments (right side).

4.3 Image classification

Figure 5 depicts the proposed classification process. This process allows classifying the state of each plant (captured as an image segment) as *water excess*, *well-watered* or *water deficit*. The optical and thermal segments obtained along the image processing

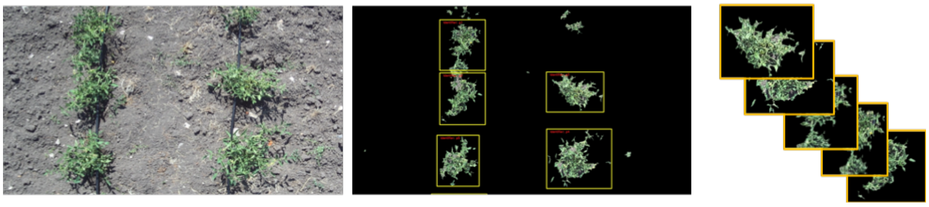


Figure 4: Segmentation of an optical image.

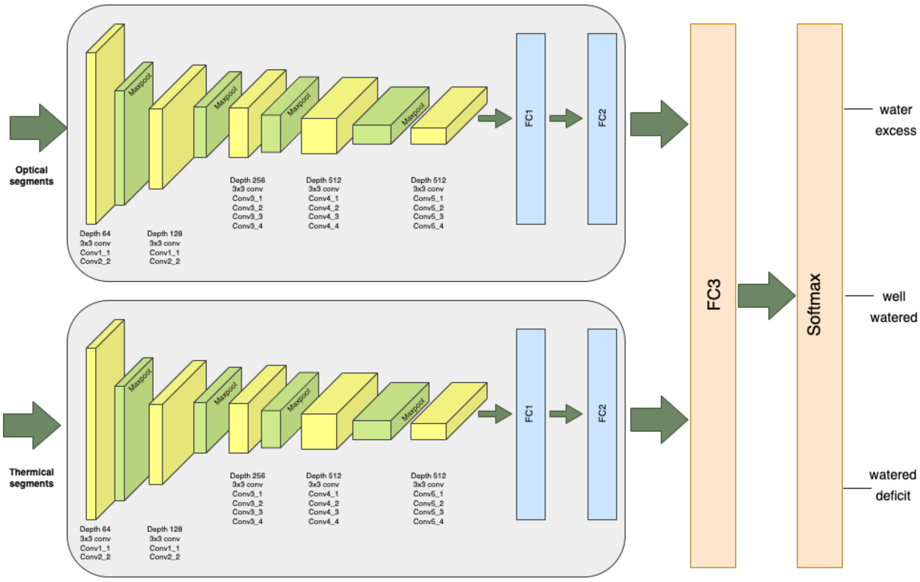

Optical segments
Maxpool
Maxpool
Maxpool
Maxpool
Depth 64 3x3 conv Conv1_1 Conv2_2
Depth 128 3x3 conv Conv1_1 Conv2_2
Depth 256 3x3 conv Conv3_1 Conv3_2 Conv3_3 Conv3_4
Depth 512 3x3 conv Conv4_1 Conv4_2 Conv4_3 Conv4_4
Depth 512 3x3 conv Conv5_1 Conv5_2 Conv5_3 Conv5_4
FC1
FC2
Thermical segments
Maxpool
Maxpool
Maxpool
Maxpool
Depth 64 3x3 conv Conv1_1 Conv2_2
Depth 128 3x3 conv Conv1_1 Conv2_2
Depth 256 3x3 conv Conv3_1 Conv3_2 Conv3_3 Conv3_4
Depth 512 3x3 conv Conv4_1 Conv4_2 Conv4_3 Conv4_4
Depth 512 3x3 conv Conv5_1 Conv5_2 Conv5_3 Conv5_4
FC1
FC2
FC3
Softmax
water excess
well watered
watered deficit

Figure 5: Classification process.

are respectively used to train two different VGG-19 classifiers. In the training step, a 10-fold cross-validation procedure [Stone, 1974] is used since it is suitable in our scenario characterized by a modest variance. Each considered VGG-19 receives a fixed size RGB image segments (optical or thermal) as input. The segments pass through a stack of convolutional layers using 3x3 filters, with a step size of one pixel. Each considered VGG has 19 layers of convolution followed by two fully connected layers (FC1, FC2) having 4096 channels each. Concerning the original VGG-19, in the adopted classifiers the last fully connected layers are retrained in order to specialize the networks to the plant classification task. Finally, the last fully connected layer (FC3) is added to learn from features extracted by both VGG channels. The output of the entire network is a softmax layer to generate probabilities for the considered classes. For the hidden layers, a rectified linear activation function ReLU function is used [Misra, 2019].

5 Evaluation

5.1 Evaluation setting

To collect the drone images, eight flights of 20 minutes have been executed. The flights are performed according to an established path (both drone operator and agronomist evaluate the correct path and the right drone distance from the field). The flight days are planned with domain experts to cover all the lifecycle of the tomato crops. According to the experts, all the flights are executed to the same day-time and with similar weather conditions to ensure that the collected images have similar characteristics. The flights are executed on a tomato field located in Pietrelcina, a small south Italy city near the province of Benevento. The UAV dataset was initially composed of 4700 thermal images and 4700 optical images. After the cleaning, filtering, processing and label tagging steps, the final dataset contained of 6600 thermal segmented images and 6600 optical segmented images. These segmented images were distributed across the classes as follows:

- 2468 normal plants (38%)
- 1723 water deficit (26%)
- 2409 water excess (36%)

making the dataset well balanced. For this reason no class imbalance reduction (CIR) techniques were applied (e.g., upsampling, downsampling, or SMOTE). The thermal image resolution is 640x510 pixels while the resolution of the optical image is 4000x3000 pixels. This final dataset is used as input to train and validate the proposed classifier. However, the dataset is split into training and test set using a 10-fold cross-validation to obtain a more reliable estimate of evaluation metrics. For the training set, the images are then labeled by the agronomist according to the classification model described in Section 4 (*water excess*, *well-watered* and *water deficit*). We evaluate the classifier performance in different conditions since the following hyperparameters are considered in the experiment:

- **Resolution size**: Different resolution sizes for thermal and optical pictures are considered: 512, 224, 448.
- **Optimization algorithm**: to optimize the neural networks, three different algorithms are evaluated in this study: the Adaptive Moment Estimation (Adam) [Kingma and Ba, 2015], the stochastic gradient descent (SGD) [Chakroun et al., 2017] and the Root Mean Square Propagation (RMSProp) [Dogo et al., 2018].
- **Batch size**: it is the number of training samples utilized in each iteration. We set the batch size to 4, 8, 16, 32.
- **Activation function**: an activation function transforms input values of a node in the output from the neuron. We used Rectified Linear Unit (ReLU) activation function [Eckle and Schmidt-Hieber, 2019]. It is a simple function that flattens the response to all negative values to zero while leaving everything unchanged for values equal or greater than zero.

- **Learning rate**: we used a rate equal to 0.0001, 1e-5;

The evaluation of the classifier performances is achieved using the following metrics: Accuracy, Precision, Recall and F1 score.
Accuracy is computed as:

$$Accuracy = \frac{Tp + Tn}{Tp + Tn + Fp + Fn} \tag{5}$$

where Tp is the number of relevant classified samples (true positive), Tn is the number of negatives that are correctly identified by the classifier (true negatives), Fn (false negatives) is the number of not retrieved relevant samples, and Fp (false positives) is the number of irrelevant retrieved samples.
The F1 score is the weighted harmonic mean of precision and recall, which, in turn, are computed according to the following formulas:

$$Precision = \frac{Tp}{Tp + Fp} \tag{6}$$

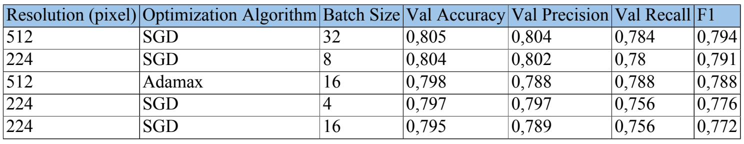
$$Recall = \frac{Tp}{Tp + Fn} \tag{7}$$

For multinomial classification, precision, recall and F1-score can be computed per-class using the same definitions reported above. The final F1-score can be evaluated, in this case, as the average of F1-scores of all the classes and is called macro-average F1 or macro-F1 for short (in this experiment where not specified, we refer to this metric).

Finally, this study also reports the validation loss values along the epochs. It is an interpretation of how well the classifier is doing. It can be computed as the sum of errors made for each sample in validation sets.

5.2 Discussion of the results

The results of the VGG-19 neural network for the best hyperparameters combinations are reported in Table 1.



Resolution (pixel)	Optimization Algorithm	Batch Size	Val Accuracy	Val Precision	Val Recall	F1
512	SGD	32	0,805	0,804	0,784	0,794
224	SGD	8	0,804	0,802	0,78	0,791
512	Adamax	16	0,798	0,788	0,788	0,788
224	SGD	4	0,797	0,797	0,756	0,776
224	SGD	16	0,795	0,789	0,756	0,772

Table 1: Performance of the best classifiers.

The table shows that the best accuracy values (0.805) are obtained when the image resolution is equal to 512 pixels, the optimization algorithm is SGD and the batch size is

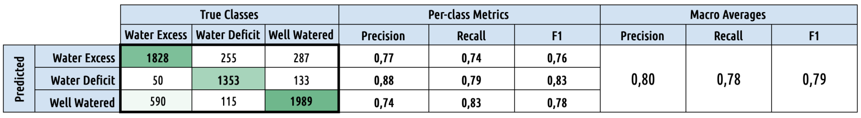
32. In all the permutations we used the ReLU activation function (for this reason it is not reported in the Table). Looking to the table, we also notice that the best optimization algorithm is SGD (only in a single case among the best five classifiers the optimization algorithm is Adamax). Figure 6 and Figure 7 also report the trend of the accuracy and the loss functions during the validation phase.

These trends show that the loss and accuracy start to be stable around after about 40 epochs (these results are also confirmed by the training accuracy and loss trends). Specifically, as Figure 7 shows, the validation accuracy reaches its maximum after 40 epochs and, using an early stop policy on this target metric with a patience parameter set up to 20 epochs the training step ends at 60 epochs.

Notice that the obtained results are lower with respect to the accuracy obtained by classifiers proposed in [Chandel et al., 2020] and [Zhuang et al., 2017]. However, there are several differences between the three studies. The classification approach described in [Chandel et al., 2020] proposes a binary classification (water stress-no stress) on different crops (maize, okra, and soybean). Both these aspects strongly influence the performance of the classifiers. However, the binary classification differs from the multinomial classification for the complexity of the faced classification issue. Similarly, the kind of crops strongly influences the classification performance (in [Chandel et al., 2020] authors show very different accuracy values according to the kind of considered culture). Finally, in [Chandel et al., 2020], the image acquisition step is performed under natural atmospheric illumination conditions at 12 noon by using a camera supported by a tripod stand. Similar considerations can also be done for the study proposed in [Zhuang et al., 2017]. However, the authors tested their classification approach on the maize plants and perform the image acquisition by using a camera installed at 4.5 meters away from the ground with a ground resolution of almost 1.5 mm. Finally, they evaluate both the accuracy of three water treatments (well-watered, reduced watered, and drought) and the accuracy of water stress (reduced watered and drought) obtaining respectively the values of 80.95 and 90.39. Also in this case we can notice that the accuracy values strongly change when a different number of treatments are considered.

To investigate the behavior of the classifier with respect to each class, we report, in Table 2, the confusion matrix obtained for the best permutation and the related per-class and aggregate performance metrics (i.e., macro averaged). As you can see, the classifier achieves similar performance for the various classes, but is slightly more efficient for detecting the state of suffering due to lack of water (which is the most important one) and a little less for detecting excess water stress. This is probably due to a greater difficulty of the network in detecting the external characteristics that the plant shows when it is irrigated too much for its needs. It should also be noted that the best recall per-class is obtained for the class of normal plants, consistent with the fact that plants not showing any of the anomalies related to stress states are easier to identify.

We also performed an ablation study to give more insights about image channels relevance. By removing optical and thermal segments in turn, the more relevant contribution to F1-score is provided by the optical segment reaching, alone, a 0.74 F1-score with respect to a 0.62 F1-score of thermal segment alone. This confirms that while optical segments remains the more important channel for water stress classification, the information provided by the thermal camera helps to improve the classification increasing



		True Classes			Per-class Metrics			Macro Averages		
		Water Excess	Water Deficit	Well Watered	Precision	Recall	F1	Precision	Recall	F1
Predicted	Water Excess	1828	255	287	0,77	0,74	0,76	0,80	0,78	0,79
	Water Deficit	50	1353	133	0,88	0,79	0,83			
	Well Watered	590	115	1989	0,74	0,83	0,78			

Table 2: Confusion matrix and per-class metrics for the best permutation.

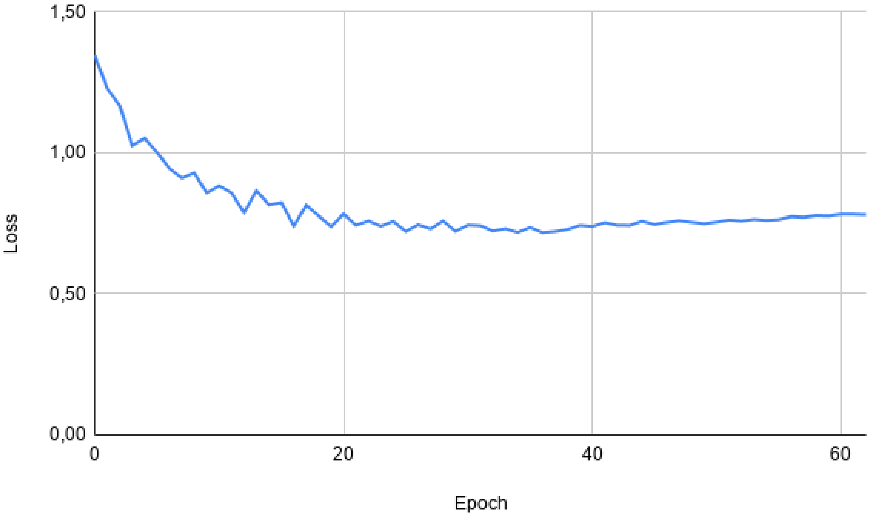


Figure 6: Loss trend for the best permutation.

the F1-score of at least a seven percent.

5.3 Threats to the validity

Different types of threats can be present in this study: construct validity, internal validity, external validity. As regards to the construct validity threats, some limitations can be due to the image acquisition process. However, different weather conditions can strongly influence the quality of drone images. To mitigate these issues several countermeasures have been adopted by planning in a very rigorous manner the flights days and paths. Moreover, all the images that have low quality, different light, and perspective conditions are delated in the cleaning step. Internal validity threats concern variables internal to our study and not considered in our experiment that could influence our observations on the dependent variable. According to this, a limitation of this study can be introduced when the datasets are labeled. However, even if the expert agronomist is involved in this step, possible errors can be introduced. To reduce this possibility, we support the agronomists in this step performing the manual insertion of the label and randomly selecting the

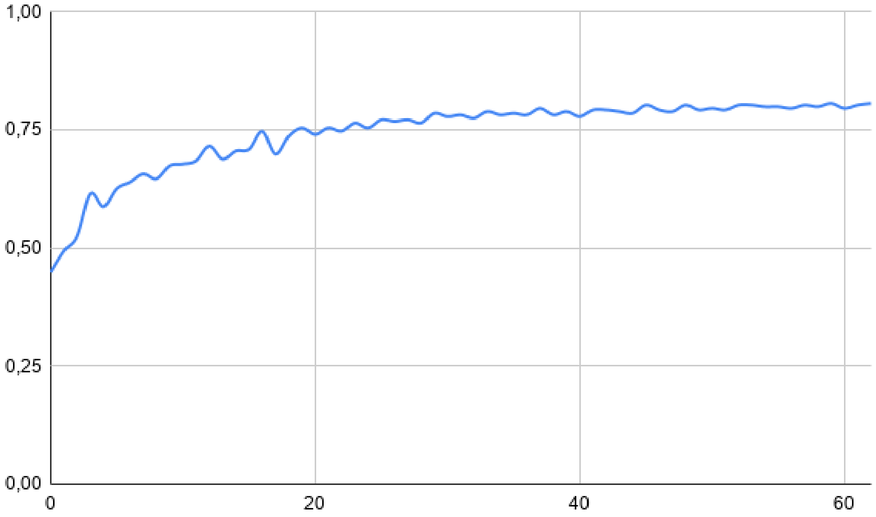


Figure 7: Accuracy trend for the best permutation.

images to evaluate (in this way we reduce the possibility that agronomist evaluation is conditioned by the plants' closeness in the field).

Finally, threats to external validity refer to the possibility to generalize the obtained outcomes. To reduce this limitation and to be more confident that results are generalizable, other fields should have experimented as future work.

6 Conclusions

This study proposes an end-to-end automatic irrigation system based on the adoption of Deep Neural Networks for the multinomial classification of tomato plants' water stress from thermal and optical aerial images. The system is implemented on a tomato field located in Benevento, a city in the south of Italy. The empirical validation of the proposed classifier is performed by capturing a large set of aerial images. These activities involved a team composed of researchers, an agronomist, the drone operator, and the field manager. The classification results are encouraging showing an accuracy of ≈ 0.80 for the best hyperparameters conditions. However, some limitations of the proposed experiment can be related to possible errors in the image labeling step since it is a manual and very onerous activity. As future work, other fields and other plants should have experimented to generalize the obtained results.

Acknowledgements

This reserach was supported by the project INNOVARE PSR Campania 2014 - 2020.

References

[Addabbo et al., 2018] Addabbo, P., Angrisano, A., Bernardi, M. L., Gagliarde, G., Mennella, A., Nisi, M., and Ullo, S. L. (2018). Uav system for photovoltaic plant inspection. *IEEE Aerospace and Electronic Systems Magazine*, 33(8):58–67.

[Agarwal et al., 2020] Agarwal, M., Singh, A., Arjaria, S., Sinha, A., and Gupta, S. (2020). Toled: Tomato leaf disease detection using convolution neural network. *Procedia Computer Science*, 167:293 – 301. International Conference on Computational Intelligence and Data Science.

[Ahmad et al., 2019] Ahmad, J., Farman, H., and Jan, Z. (2019). *Deep Learning Methods and Applications*, pages 31–42. Springer Singapore, Singapore.

[Al-Ghobari and Mohammad, 2011] Al-Ghobari, H. M. and Mohammad, F. S. (2011). Intelligent irrigation performance: evaluation and quantifying its ability for conserving water in arid region. *Applied Water Science*, 1(3):73–83.

[Ardimento et al., 2021] Ardimento, P., Aversano, L., Bernardi, M. L., and Cimitile, M. (2021). Deep neural networks ensemble for lung nodule detection on chest CT scans. In *International Joint Conference on Neural Networks, IJCNN 2021, Shenzhen, China, July 18-22, 2021*, pages 1–8. IEEE.

[Aversano et al., 2020] Aversano, L., Bernardi, M. L., Cimitile, M., Iammarino, M., and Rondinella, S. (2020). Tomato diseases classification based on vgg and transfer learning. In *2020 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor)*, pages 129–133.

[Brahimi et al., 2017] Brahimi, M., Boukhalfa, K., and Moussaoui, A. (2017). Deep learning for tomato diseases: Classification and symptoms visualization. *Appl. Artif. Intell.*, 31(4):299–315.

[Chakroun et al., 2017] Chakroun, I., Haber, T., and Ashby, T. J. (2017). Sw-sgd: The sliding window stochastic gradient descent algorithm. *Procedia Computer Science*, 108:2318–2322. International Conference on Computational Science, ICCS 2017, 12-14 June 2017, Zurich, Switzerland.

[Chandel et al., 2020] Chandel, N. S., Chakraborty, S. K., Rajwade, Y. A., Dubey, K., Tiwari, M. K., and Jat, D. (2020). Identifying crop water stress using deep learning models. *Neural Computing and Applications*.

[Deng et al., 2014] Deng, L., Yu, D., et al. (2014). Deep learning: methods and applications. *Foundations and Trends® in Signal Processing*, 7(3–4):197–387.

[Dogo et al., 2018] Dogo, E. M., Afolabi, O. J., Nwulu, N. I., Twala, B., and Aigbavboa, C. O. (2018). A comparative analysis of gradient descent-based optimization algorithms on convolutional neural networks. In *2018 International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS)*, pages 92–99.

[Eckle and Schmidt-Hieber, 2019] Eckle, K. and Schmidt-Hieber, J. (2019). A comparison of deep networks with relu activation function and linear spline-type methods. *Neural Networks*, 110:232–242.

[Feng Wang et al., 2011] Feng Wang, Taisheng Du, Rangjian Qiu, and Pingguo Dong (2011). Effects of water stress at different growth stage on greenhouse multiple-trusses tomato yield and quality. In *2011 International Conference on New Technology of Agricultural*, pages 282–287.

[Ferrández-Pastor et al., 2018] Ferrández-Pastor, F. J., García-Chamizo, J. M., Nieto-Hidalgo, M., and Mora-Martínez, J. (2018). Precision agriculture design method using a distributed computing architecture on internet of things context. *Sensors*, 18(6).

[Gallardo et al., 2020] Gallardo, M., Elia, A., and Thompson, R. B. (2020). Decision support systems and models for aiding irrigation and nutrient management of vegetable crops. *Agricultural Water Management*, 240:106209.

[Gao et al., 2020] Gao, Z., Zhonhwei, L., Zhang, W., Lv, Z., and Xu, Y. (2020). Deep learning application in plant stress imaging: A review. *AgriEngineering*, 2:430–446.

[Hoffmann et al., 2016] Hoffmann, H., Jensen, R., Thomsen, A., Nieto, H., Rasmussen, J., and Friborg, T. (2016). Crop water stress maps for an entire growing season from visible and thermal uav imagery. *Biogeosciences*, 13(24):6545–6563.

[Kawasaki et al., 2015] Kawasaki, Y., Uga, H., Kagiwada, S., and Iyatomi, H. (2015). Basic study of automated diagnosis of viral plant diseases using convolutional neural networks. In Bebis, G., Boyle, R., Parvin, B., Koracin, D., Pavlidis, I., Feris, R., McGraw, T., Elendt, M., Kopper, R., Ragan, E., Ye, Z., and Weber, G., editors, *Advances in Visual Computing*, pages 638–645, Cham. Springer International Publishing.

[Kingma and Ba, 2015] Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In Bengio, Y. and LeCun, Y., editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

[Lima et al., 2016] Lima, R., GarcÃa-Tejero, I., Lopes, T., Costa, J., Vaz, M., DurÃ¡n-Zuazo, V., Chaves, M., Glenn, D., and Campostrini, E. (2016). Linking thermal imaging to physiological indicators in carica papaya l. under different watering regimes. *Agricultural Water Management*, 164:148–157. Enhancing plant water use efficiency to meet future food production.

[Lu et al., 2017] Lu, H., Cao, Z., Xiao, Y., Zhuang, B., and Shen, C. (2017). Tasselnet: counting maize tassels in the wild via local counts regression network. *Plant Methods*, 13(1):79.

[Marcu et al., 2019] Marcu, I. M., Suciu, G., Balaceanu, C. M., and Banaru, A. (2019). Iot based system for smart agriculture. In *2019 11th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, pages 1–4.

[Miras-Avalos et al., 2019] Miras-Avalos, J., Rubio-Asensio, J. S., Ramírez Cuesta, J., Maestre, J., and Intrigliolo, D. (2019). Irrigation-advisor—a decision support system for irrigation of vegetable crops. *Water*, 11:2245.

[Misra, 2019] Misra, D. (arXiv pre-print, 2019). Mish: A self regularized non-monotonic neural activation function.

[Nendel, 2014] Nendel, C. (2014). *MONICA: A Simulation Model for Nitrogen and Carbon Dynamics in Agro-Ecosystems*, pages 389–405. Springer International Publishing, Cham.

[Petrie et al., 2019] Petrie, P. R., Wang, Y., Liu, S., Lam, S., Whitty, M. A., and Skewes, M. A. (2019). The accuracy and utility of a low cost thermal camera and smartphone-based system to assess grapevine water status. *Biosystems Engineering*, 179:126–139.

[Rao and Sridhar, 2018] Rao, R. N. and Sridhar, B. (2018). Iot based smart crop-field monitoring and automation irrigation system. In *2018 2nd International Conference on Inventive Systems and Control (ICISC)*, pages 478–483.

[Rinaldi and He, 2014] Rinaldi, M. and He, Z. (2014). Chapter six - decision support systems to manage irrigation in agriculture*present address: Consiglio per la ricerca e la sperimentazione in agricoltura, cereal research centre, s.s. 673km 25,200, 71122 foggia, italy. volume 123 of *Advances in Agronomy*, pages 229 – 279. Academic Press.

[Saraf and Gawali, 2017] Saraf, S. B. and Gawali, D. H. (2017). Iot based smart irrigation monitoring and controlling system. In *2017 2nd IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT)*, pages 815–819.

[Simonyan and Zisserman, 2014] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition.

[Singh et al., 2018] Singh, A., Ganapathysubramanian, B., Sarkar, S., and Singh, A. (2018). Deep learning for plant stress phenotyping: Trends and future perspectives. *Trends in plant science*, 23 10:883–898.

[Stone, 1974] Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *Roy. Stat. Soc.*, 36:111–147.

[Torres-Sanchez et al., 2020] Torres-Sanchez, R., Navarro-Hellin, H., Guillamon-Frutos, A., San-Segundo, R., Ruiz-Abellón, M. C., and Domingo-Miguel, R. (2020). A decision support system for irrigation management: Analysis and implementation of different learning techniques. *Water*, 12(2).

[Tseng et al., 2018] Tseng, D., Wang, D., Chen, C., Miller, L., Song, W., Viers, J., Vougioukas, S., Carpin, S., Ojea, J. A., and Goldberg, K. (2018). Towards automating precision irrigation: Deep learning to infer local soil moisture conditions from synthetic aerial agricultural images. In *2018 IEEE 14th International Conference on Automation Science and Engineering (CASE)*, pages 284–291.

[van Diepen et al., 1989] van Diepen, C., Wolf, J., van Keulen, H., and Rappoldt, C. (1989). Wofost: a simulation model of crop production. *Soil Use and Management*, 5(1):16–24.

[Van Hulle et al., 2016] Van Hulle, M., Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., and Stefanovic, D. (2016). Deep neural networks based recognition of plant diseases by leaf image classification. *Computational Intelligence and Neuroscience*, 2016:3289801.

[Vij et al., 2020] Vij, A., Vijendra, S., Jain, A., Bajaj, S., Bassi, A., and Sharma, A. (2020). Iot and machine learning approaches for automation of farm irrigation system. *Procedia Computer Science*, 167:1250–1257. International Conference on Computational Intelligence and Data Science.

[Xu et al., 2018] Xu, R., Li, C., Paterson, A. H., Jiang, Y., Sun, S., and Robertson, J. S. (2018). Aerial images and convolutional neural network for cotton bloom detection. *Frontiers in Plant Science*, 8:2235.

[Yu and Shi, 2003] Yu and Shi (2003). Multiclass spectral clustering. In *Proceedings Ninth IEEE International Conference on Computer Vision*, pages 313–319 vol.1.

[Zhuang et al., 2017] Zhuang, S., Wang, P., Jiang, B., Li, M., and Gong, Z. (2017). Early detection of water stress in maize based on digital images. *Computers and Electronics in Agriculture*, 140:461 – 468.