UNIVERSITY
INSTITUTE
OF LISBON

Thermal Image Analysis for Precision Agriculture

Master in Integrated Decision Support Systems

André Simões Novo

Supervisor:
Professor Octavian Adrian Postolache, Professor Catedrático
Iscte – Instituto Universitário de Lisboa

Supervisor:
Professor Joaquim Gabriel Magalhães Mendes, Professor Catedrático
FEUP – Faculdade de Engenharia da Universidade do Porto

September, 2025

TECNOLOGIAS
E ARQUITETURA

Department of Information Science and Technology

Thermal Image Analysis for Precision Agriculture

Master in Integrated Decision Support Systems

André Simões Novo

Supervisor:
Professor Octavian Adrian Postolache, Professor Catedrático
Iscte – Instituto Universitário de Lisboa

Supervisor:
Professor Joaquim Gabriel Magalhães Mendes, Professor Catedrático
FEUP – Faculdade de Engenharia da Universidade do Porto

September, 2025

[This page is intentionally left blank.]

Acknowledgment

I would like to thank my supervisors Professor Octavian Postolache from Iscte-IUL and Professor Joaquim Mendes from FEUP for providing me the equipment and supervision, Professor Mariana Mota from ISA, Professor Elsa Cardoso and my family.

A special thanks go to the instituitions that support my work during the master program Iscte-Instituto universitário de Lisboa and Instituto de Telecomunicações, Lisboa Portugal and "2025 Third International Conference on Industry 4.0 Technology (I4Tech)"

[This page is intentionally left blank.]

Resumo

O crescimento da população mundial exige uma melhor gestão dos recursos naturais, nomeadamente uma utilização mais sustentável da água. Este projeto propõe então uma solução baseada em técnicas de aprendizagem profunda que recorre a imagens térmicas, utilizando aos seus valores térmicos verdadeiros para realizar uma deteção e previsão não invasiva do stress hídrico das plantas agrícolas mais preciso.

Quando integrado num sistema aéreo não tripulado (drone), esta abordagem tem potencial para monitorizar campos agrícolas extensos.

O método proposto visa a reduzir o uso de técnicas invasivos, de maior tempo de implementação e que necessitem de mais mão de obra como a instalação de sensores terrestres ao longo do campo agrícola. Ao substituir estes métodos por um sistema aéreo, torna-se possível obter medições mais rápidas, flexíveis a menor custo.

A metodologia desenvolvida está estruturada em duas etapas principais: (i) a deteção e localização das plantas através de um detetor de objetos YOLO; e (ii) a classificação do nível de stress hídrico por meio de uma rede neuronal convolucional (CNN), a qual utiliza diretamente os valores térmicos reais presentes nas imagens. Os dados utilizados para o treino dos modelos foram obtidos a partir de três conjuntos de plantas de salsa submetidas a diferentes níveis de hidratação do solo — 25%, 50% e 85% — determinados através de métodos de pesagem e sensores de humidade do solo.

O modelo de deteção alcançou um mAP@50 de 99,5% e um mAP@50-95 de 91.5%. A CNN apresentou uma precisão de 99,05% na classificação do stress hídrico utilizando a aprendizagem por transferência e 84,97% ao utilizar um modelo personalizado.

Esta abordagem prova que as imagens térmicas são capazes de realizar uma avaliação rápida e precisa das necessidades hídricas das plantas. Ao disponibilizar esta informação de forma prática e ágil torna possivél apoiar os agricultores na irrigação das suas culturas de forma mais precisa e eficiente, poupando água no processo.

Palavras chaves: Imagem térmica, Stress hídrico em plantas, Aprendizagem profunda, Agricultura de precisão, Detecção de objectos, Redes neuronais convolucionais

[This page is intentionally left blank.]

Abstract

The growth of the world population requires better management of natural resources, particularly more sustainable use of water. This project therefore proposes a solution based on deep learning techniques that uses thermal images making use of their true thermal values to perform a more accurate non-invasive detection and prediction of water stress in agricultural plants.

When integrated into an unmanned aerial system (drone), this approach has the potential to monitor large agricultural fields.

The proposed method aims to reduce the use of invasive techniques that take longer to implement and require more manual labor, such as the installation of ground sensors throughout the agricultural field. By replacing these methods with an aerial system, it becomes possible to obtain faster, more flexible measurements at a lower cost.

The methodology developed is structured in two main stages: (i) the detection and location of plants using a YOLO object detector; and (ii) the classification of the level of water stress using a convolutional neural network (CNN), which directly uses the actual thermal values present in the images. The data used to train the models were obtained from three sets of parsley plants subjected to different levels of soil moisture—25%, 50%, and 85%—determined using weighing methods and soil moisture sensors.

The detection model achieved a mAP@50 of 99.5% and a mAP@50-95 of 91.5%. The CNN showed an accuracy of 99.05% in water stress classification using transfer learning and 84.97% when using a custom model.

This approach proves that thermal images are capable of performing a quick and accurate assessment of plants' water needs and by making this information available in a practical and agile way, it is possible to support farmers irrigating their crops more accurately and efficiently, saving water in the process.

Keywords: Thermal Imaging, Plant water stress, Deep learning, Precision agriculture, Object detection, Convolutional Neural Networks

[This page is intentionally left blank.]

Contents

List of Figures

List of Tables

[This page is intentionally left blank.]

List of Acronyms

AI: Artificial Intelligence
AP: Average Precision
CLAHE: Contrast-Limited Adaptive Histogram Equalization
CNN: Convolutional Neural Networks
COCO: Common Objects in Context (dataset)
CWSI: Crop Water Stress Index
DenseNet: Densely Connected Convolutional Network
EfficientDet: Efficient Object Detector
FLIR: Forward Looking Infrared
GPU: Graphics Processing Unit
ICE: Image Contrast Enhancement
IoU: Intersection over Union
JPG: Joint Photographic Experts Group
kNN: k-Nearest Neighbors
LeNet5: LeNet-5 Convolutional Neural Network
MLP: Multi-Layer Perceptron
mAP: Mean Average Precision
MobileNet: Lightweight CNN for mobile vision
MobileNetV2: Second version of MobileNet, optimized for speed and accuracy
P: Precision
PRISMA: Preferred Reporting Items for Systematic reviews and Meta-Analyses
R: Recall
RCNN / R-CNN: Region-based Convolutional Neural Network
ResNet: Residual Neural Network
RetinaNet: CNN-based object detector
RGB: Red, Green, and Blue
SSD: Single Shot MultiBox Detector
SVM: Support Vector Machine
UAV: Unmanned Aerial Vehicle
U-Net: Convolutional Network for image segmentation
VGG: Visual Geometry Group Network (e.g., VGG16, VGG19)
YOLO: You Only Look Once
mAP@50: Mean Average Precision at 50% Intersection over Union threshold
mAP@50:95: Mean Average Precision averaged over IoU thresholds from 50% to 95%

[This page is intentionally left blank.]

CHAPTER 1

Introduction

1.1. Context and objectives

Water is an essential resource and with the rapid rise of population growth, its total amount will tend to diminish since it is a limited resource. This problem with the fact that and drought is one of the most harmful abiotic factors to plants [4] leads to the necessity of optimizing its consumption not only to save water but also to maximize the crop yields [3,4].

This growth was an incentive to use of Artificial Intelligence (A.I) technologies in smart farming to monitor the crops growth through a collection of data about crop conditions and environments to form datasets in order to train the A.I models [10].

Thermography is a solution that has been used to monitor the water status of various plants, its a non-invasive and non-destructive agricultural practice that supports the growing demand for efficiency in food production and water control[2].

This technique allows the farmers to monitor soil salinity stress, crop diseases, yield estimation, maturity evaluation, bruise detection and water stress in plants due to the different types of radiation emoted by these factors making thermal imaging a relevant addition to the field of agriculture that does not damage or provoke changes in the soil unlike traditional soil and plant based methods, possesses swift implementation in the fields and fast and accurate monitoring of the crops[3,4].

Thermographic images enable the extraction of detailed radiometric measurements related to temperature fluctuations, which result from the thermal radiation emitted by plant leaves. During water stress, stomatal closure reduces transpiration, causing leaf temperature to rise. The greater the stress level, the more the stomata close, and the higher the resulting temperature.

This physiological response allows the differentiation between stressed and non-stressed plants through thermal imaging [3,4].

These images can then be used with deep learning techniques such as the YOLO [28], EfficientDet [31], for the purpose of detection and classification of objects or convolutional neural networks like the VGG-19[21] to simply classify the image based on the identification of thermal patterns, these patterns can also be used to differentiate patients with Covid19 [19], detect of pedestrians [32] or to both detect and classify pedestrians, dogs and cars for example [30].

The pipeline is structured in two main stage: (i) the detection and location of plants using a YOLO object detector; and (ii) the classification of the level of water stress using a convolutional neural network (CNN), which uses the actual thermal values present in the images. These images are normalized by subtracting the room temperature from their thermal readings, enabling the model to make consistent predictions across different days.

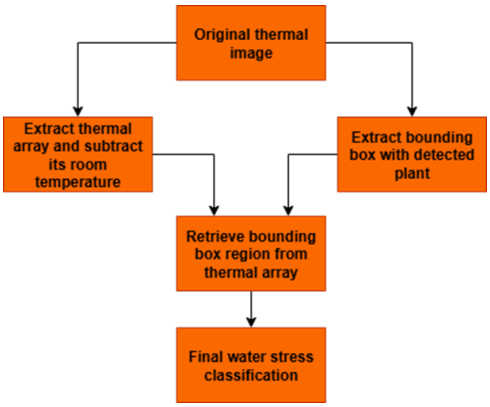


FIGURE 1.1. Prediction Pipeline

If this methodology is integrated in a UAV with a thermal camera, the used approach provides a non-invasive and agile way to quickly assess water stress, promoting the replacement of ground soil moisture sensors.

To create our training dataset we will be using a E60 thermal camera to take pictures of 3 plants of parsley (Petroselinum crispum) subjected to three distinct soil moisture regimes: 25%, 50%, and 85% of the soil's water holding capacity.

These target percentages were precisely established and maintained using the gravimetric method, whereby pot weights were calibrated against their saturated and oven-dry states, with the values subsequently regulated and verified using a calibrated soil moisture sensor to ensure accuracy. All imaging was conducted under controlled environmental conditions using a FLIR E60 thermal camera following a standardized protocol to minimize external variability.

The parsley plant was chosen due to the fact that it is an agricultural plant grown for food commerce purposes and of easy access.

An article about this project was also published under "2025 Third International Conference on Industry 4.0 Technology (I4Tech)".

1.2. Motivation

There are different approaches for monitoring crop water stress, usually falling into soil-based, plant-based, and atmosphere-based measurements. These approaches usually take advantage of traditional tools such as soil moisture sensors, pressure chambers, and

leaf diffusion porometers which can detect stress but are often labor-intensive and difficult to use effectively in field conditions as it was previously mentioned.

Thermal imaging, on the other hand, provides a practical, non-invasive, and efficient alternative, allowing plant water status to be assessed with minimal manual effort and without disturbing the crop [1]. This project aims to assist in the development of a deep learning methodology pipeline that utilizes both the rgb part of the thermal image and the real thermal values in order to efficiently assess water stress in food crops.

The proposed methodology if integrated with aerial unmanned drones, provides farmers with a rapid and agile method for assessing water stress in their crops. Besides implementation speed, a single drone should be able to efficiently survey large agricultural areas from above, replacing other traditional methods that as mentioned before require more implementation time, are more labor intensive and require more hardware installations and maintenance.

Furthermore, a drone-based system offers scalability, since as a farm expands the same drone can simply adjust its flight paths to cover new plots of land, whereas ground sensors would require additional units.

These characteristics position thermal imaging as a cost-effective and practical approach for monitoring water stress, particularly in medium to large agricultural operations. Its rapid deployment, minimal hardware requirements and scalability make it an attractive alternative to conventional ground-based sensing methods.

1.3. Research questions

- How can we take advantage of the embedded thermal data within the images to upgrade our predictive models?

  Since thermal images possess both their visual thermal data and numeric thermal data embedded within the image itself, as such, if we are able to feed the model with this data and not only the simple thermal RGB data, we are able to predict different types of thermal images not just for example grey scale images and on different scales.

- What kind of models and pre-processing should we use in order to detect and classify stressed plants?

  Thermal images are different than normal images, not only because the scale of the image is importance to have the correct visualization but since these kind

of images also possess thermal values embedded some of the more common methods might not work for they might delete this information.

- What sort of experiment should we conduct in order to create a quality thermal image dataset, with different amounts of water stress?

  Here we wondered what sort of equipment and techniques we were going to use in order to conduct an experiment to prepare the dataset to train the models.

CHAPTER 2

Literature Review

In order to gain insights about the topics at hand (plant water stress and deep learning applied to thermal images) a research in the SCOPUS database was made using the PRISMA methodology.

There were a total of two refined queries, one for researching the effects of water stress in plants and the route the authors took to perform the experiment successfully. The other query was meant to research how can deep learning be applied to thermal images on FLIR datasets (this was filtered only to FLIR datasets because some thermal images from other brands are sometimes subjected to other treatments related to licenses for us to be able to get the thermal metadata).

Both queries were filtered to contain only articles and conference papers that were in English and there were no duplicates between them, after that each branched into their domain.

The research on deep learning in thermal imagery was focused solely on image classification, segmentation and object detection excluding generative models in order to learn which type of architecture is the most popular and the best. The database was filtered between 2020 and above since its the year YOLOv4 was released which outperforms YOLOv3 by a large margin as it was mentioned in [28], during the same year YOLOv5 was also released, and this way we are able to obtain information on only the latest models.

The process then advanced into the initial screening of the papers where the title and abstract of the papers were analyzed, where a rule was applied for each group of articles. In the modeling research there was a focus on articles that had the techniques that were sought out to be used in this paper (CNN and object detection).

For the plant water stress query, the results were filtered to appear only between 2018 and above and were confined only to the agricultural area. On the research about the water stress in plants the selected articles talked about splitting the plants into different groups each with different irrigation experiments and whose focus was the water stress and its effect on the leafs of the plants. There was a total of 143 articles and it was filtered to 78, then we tried to access each of the filtered documents and were left with 59 total documents to perform a deep analysis.

After the entire screening from PRISMA we were left with 37 articles, we can see the prisma results on figure 2.1.

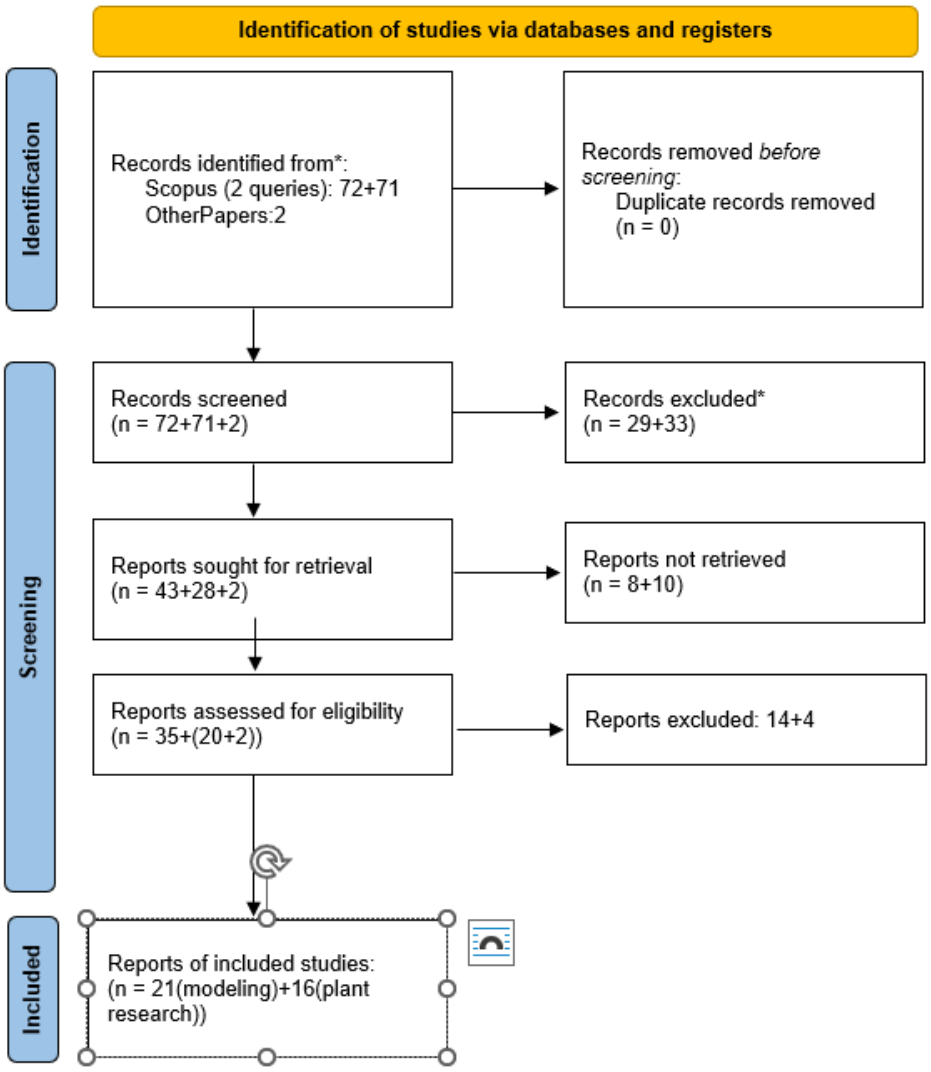


FIGURE 2.1. Prisma Diagram

2.1. Water stress and its effects on plants and their thermal image research

Query used: TITLE-ABS-KEY (("Thermography" OR "Thermal image") AND ("Drought" OR "Water stress" OR "Hydric stress") AND NOT ("Real time" OR "UAV" OR "aerial")) AND PUBYEAR > 2018 AND PUBYEAR < 2026 AND (LIMIT-TO (SUBJAREA , "AGRI")) AND (LIMIT-TO (DOCTYPE , "ar") OR LIMIT-TO (DOCTYPE , "cp")) AND (LIMIT-TO (LANGUAGE , "English"))

In order to create a dataset and analyze the thermal images, it was needed information within the field of horticulture, which is the agriculture branch that studies the different ways to maximize what plants can produce and minimize wasted resources. For this thesis in particular, the research is about the effects of water stress on plants and their thermal image.

Plants have two main physiological processes, photosynthesis, where the light is converted into chemical energy that can later be released to fuel the necessities of the organism, in more detail it is the synthesis of carbohydrates from carbon dioxide and water with the generation of oxygen. And transpiration where the plant will dissipate heat at the exchange of water through leaf surfaces containing pores known as the stomata in order to regulate its own temperature.[1]

A plants response to the effects of droughts has three stages. Stage 1(mild water stress), where there is a reduction of the transpiration caused by a decline of stomatal conductance, which will cause the stomata to close reducing the plants transpiration. Stage 2 (moderate water stress): the stomatal conductance is further decreased, and a significant decrease in the photosynthetic activity of the plant. Finally stage 3(severe water stress), where the stomatal conductance drops below its threshold value, impairing the photosynthetic capacity of the plant and causing permanent damage to its photosystems [1].

Water stress can sometimes be observed through the wilting of the plant without the need of tools, other times it might be under stress which is hindering its growth but there are no visible clues of it. Or through the usage of more precise methods like soil moisture sensors, pressure chambers and leaf diffusion porometers are very popular when it comes to assessing the water stress of the plants. Although popular these methods are labor intensive and difficult to be carried out under field conditions. That is where the thermal image comes in, it is a non-destructive tool that can distinguish levels of water stress that we cannot using simply our eyes cannot, are easier to implement and require less manual labor[1,2].

Different types of plants show different reactions when under water stress, according to this paper [3] for example, the leaf temperature of the tree *Pometia tomentose* is of 31.8ºC while in the *Mezzettipsis creaghii* it was of 21.6ºC. Common beans and buckwheat are more sensible to water stress than amaranth [12], this reinforces the idea that each type of plant have possess different tolerances to water stress, even simply changing its genotype might make the plant more resistant to water stress as it is the example of the strawberry [13], the cotton plant [14], mungbean and chickpeas [15].

On the experiment mentioned by [3], the authors irrigated the plants to 100% of the field capacity and then 75%, 50% and 25% of full irrigation treatment was applied respectively to each plant. They were grown in a lysimeter-type pot. In order to take the pictures the authors used a FLIR E53 and FLIR TOOLS for the analysis of the image.

The results of these experiment demonstrated that maize plants under severe stress showed temperatures between 20.9ºC and 29.3ºC, those under moderate stress achieved temperatures of 20.1ºC to 27.8ºC, mild temperatures of 19.5ºC and 27.4ºC and finally those with full irrigation had temperatures between 17.5ºC and 23.3ºC from march 8

to april 25. The highest spike temperature values on april 25th were I25 (30.8°C), I50 (29.1°C), and I75 (29.6°C) and I100(26.91°C). As we can see the minimum as well as the maximum temperature between the heavy and medium stress show notable differences when compared to the plot with no hydric stress.

Another study by Han[4], suggested that the standard deviation of the canopy temperature of the maize which is , was able to categorize 3 different types of stress: higher than 3ºC is the equivalent to high water stress, between 2 ºC and 3ºC is medium water stress and under 2ºC is no water stress. Continuing with the analysis of the paper, the author did his experiment in a Mediterranean climate with hot dry summers and mild wet winters, this temperature has an average of 18.8ºC with a lowest average of 10ºC and highest average of 28ºC. The experiment was conducted on the wheat plant throughout the growing season, they were planted on lysimeter-type pots and had 4 different irrigation levels, 100% soil capacity, 70% to 80%, 45% to 55% and finally 20% to 30%.

The next experiment [5] used different watering regimes on melon plants, IR1- 120%, IR2 100%, IR3- 80%, IR4 60%, and IR5- 40% of its own evapotranspiration. The soil moisture was determined by collecting samples and using the gravimetric method, the leaf water potential was measured using a pressure camera, both of these measurements were taken every 15 days for 60 days. The evapotranspiration was estimated by direct readings of daily evaporation in Class A pan, the Canopy temperature was measured with a FLIR E40 thermal camera and the air temperature with a temperature sensor.

The authors then tried relating the canopy temperature with the air temperature by doing the difference between each other; this metric was used due to its feasibility with thermographic images. Positive values in this difference mean that the plant has lower transpiration rates, implying it is under stress; this difference shows high negative correlation with soil moisture, so the lower the value, the more ideal are the soil moisture conditions. Without mulching, the temperatures ranged from -3 degrees to +3.8 degrees celcius [5].

In this paper[6] the leaf temperature is used as one of the indicators of drought, increased temperature equals higher water stress, they started with seeds and after 8 of them were planted and the soil was kept them well watered in 5L pots.

After day 14 they were split to 2 plants per pot. Until day 25 (which was the day the plants fully opened their trifoliate leaves) they were still well watered and after that, the plants were induced into a state of drought by suspending the irrigation until it achieved a field capacity of 20%. After 3 days they plant achieved a soil moisture of 50% field capacity and after 6 days it achieved 20%.

In order to control the field moisture, firstly the soil was air dried and weighted to obtain its dry weight and a moisture sensor to obtain the dry weight reading, after that

it was saturated and left to drain to obtain its wet weight and once again the sensor was used to get the wet weight reading, the difference between these two was the pot water holding capacity. This data was used to check the amount of water to be replenished in irrigation.

This experiment was done on two variations of soybeans (Don Mario and Mamoré), and upon analysing the temperature of the plants with no EBL concentrations we see a huge difference of up to 7 degrees celcius between each plant on the various stress levels. The authors concluded that EBL has a positive impact in the plants water regulation, increasing its water use efficiency.

In the experiment presented in [7], the authors also used similar methods, but instead of using the soil moisture index (SMI) the authors decided to use the stem water potential with a Scholander pressure chamber during the hottest hours of the day, watering the plants and maintaining them above a certain threshold to attain no, medium and high water stress levels, by replenishing 100%, 50% and 25% of the daily water usage.

The authors compared the crop water stress index with the stomatal conductance index in order to classify the water status in the plant and reached the conclusion that the crop water stress index was a more reliable indicator. This metric is calculated by using the canopy temperature, wet temperature which was achieved by coating the leaf with water, the dry temperature which is achieved by coating the leaf with Vaseline. These values can differ from 0 to 1, 0 meaning the plant is not under stress and 1 meaning it is under severe stress, we can see the formula on the next function (1).

$$(1) \quad \text{CWSI} = \frac{T_{\text{canopy}} - T_{\text{wet}}}{T_{\text{dry}} - T_{\text{wet}}}$$

The following paper[8] explains the water stress effects on wheat during the winter using thermal. These images were the primary method used to quickly capture canopy temperature data. To achieve these results the soil was kept under different irrigation treatments, adequate irrigation, moderate, slight deficit and severe deficit.

While the yield of the maize plant with moderate irrigation did not change too much when compared to the plant at full field capacity, the plant with a slight water deficit and especially the plant with severe water deficit did lose quite a bit, producing half the yield of the fully irrigated plant.

Analysis of leaf relative water content (RWC) revealed limited variation across all irrigation treatments at the jointing stage in both years. However, during the flowering and grain-filling stages, a clear divergence emerged. A slight decrease in RWC was observed in plants under mild water deficit compared to the irrigated and moderately irrigated groups. In contrast, plants subjected to severe water deficit exhibited a pronounced reduction in RWC across all growth stages.

Stomatal conductance mirrored this trend, showing a significant drop in the deficit treatments compared to the fully and moderately irrigated plants, particularly during the flowering and filling stages. Transpiration rates closely followed the pattern of stomatal conductance.

Photosynthetic performance increased substantially during the reproductive stages (flowering and filling) in the irrigated and moderately watered plants. Conversely, photosynthesis remained consistently low in the severe deficit treatment and showed no significant improvement in the mild deficit group. These physiological measurements collectively demonstrated a strong negative correlation with the Crop Water Stress Index (CWSI).

Although thermal images can be an accurate diagnosis of the water status of a plant, they are affected by external conditions such as solar radiation, air temperature, wind speed and humidity which will affect the accuracy of the thermal readings, the CWSI aims to minimize these variations. The CWSI value is closely related to the to leaf water potential, stomata resistance and soil water[8].

Direct solar radiation will increase the leaves temperature, causing it to transpire more than shaded leaves, this will reinforce fact that solar radiation will have an impact on thermal readings[4]

Although CWSI gets good accuracy when identifying the water stress of plants as it establishes a relation between canopy-to-air temperature difference and vapor pressure deficit, this is a method requires site-specific calibration, and reference surfaces (wet and dry areas), which is hard for farmers to set up[4, 8].

Gutíerrez-Gordillo [9] mentions that the difference between the canopy and air temperature is widely used due to its simple approach in order to normalize the absolute canopy temperature, but measurements such as the CWSI that implements non-water-stress baselines (NWSB) and water-stress baselines (WSB) are more resilient to external conditions.

In the paper [10], the results showed that leaf temperature increased under water deficit conditions due to stomatal closure, and CWSI values effectively quantified stress levels, rising to 0.76 under severe stress and dropping after irrigation. The study concluded that CWSI, derived from thermal imaging, is a reliable tool.

The higher the water stress, not only will the plant temperature be higher, the more the plant will suffer its effects for example withering and lower yield.

In this experiment with the tomato, although the total yields of the plant with irrigation at 75% are lower than the plant a irrigation of 100% as it gives less fruits per plant, it has a higher percentage of marketable plants when compared. Still the plant at 100% irrigation will still give more marketable fruits due to higher overall production (14% to 19% more fruits produced both in 2018 and 2019 while the difference in marketable products

is only 4% to 3%) the plant at 75% possesses a higher concentration of Lycopene which means the tomato is of higher quality and saves more water in the process. In short the higher the water stress the higher the amount of lycopene, but the challenge is balancing these values so that we can get a good yield and still produce high quality plants.

To measure the water stress levels it was used the CWSI and coated the some leaves with water once again the to obtain the non-stress reference and other leaves with petroleum to obtain the stressed reference. The images were taken from a vertical angle to remove the effect of shadows.

As for the irrigation treatment, the authors irrigated according to the evapotranspiration level of the plants (100%, 75% and 50%).[11].

On paper [12], results showed that amaranth and quinoa maintained higher photosynthesis and yield under water stress, while common bean was highly sensitive, suffering significant reductions in gas exchange and productivity. Canopy temperature increased with water stress and showed strong negative correlations with yield and physiological performance across all species.

This study [13] evaluated the drought tolerance of four strawberry genotypes (Rubygem, Festival, 33, and 59) under well-watered (IR100) and water-stressed (IR50) conditions in a high-tunnel Mediterranean environment. The research found that water stress significantly reduced yield and photosynthetic parameters but increased the CWSI and irrigation water use efficiency. Genotype 59 was identified as the most drought-tolerant, exhibiting the highest yield and highest water use efficiency, lowest CWSI, and superior photosynthetic performance under stress.

The treatment in the experiment with cotton was different, the authors watered the soil and then after 105 days post sowing they left 2 random genotypes under no irrigation for the rest of the experiment while another was kept under a well watered regime, after 4 days the plants were considered under mild drought and after 14 days under severe drought, all the data was collected on those two days and the experiment was conducted on a greenhouse with controlled environmental conditions[14].

In the experiment reported on article [15], the authors planted different types of beans in 20x20cm pots filled with 15kg of soil (red soil for mungbeans and black soil for chickpeas).

These plants had well watered periods and water stress treatment like the experiments mentioned before, but this time in order to measure the moisture of the soil, a small sample was taken out of the pot using aluminium boxes at 10cm and 15cm depth, weighted and then oven-dried for 72h at 105 degrees celsius to determine the soil moisture content.

The images were taken over 8 days on every alternative day after 40 days of sowing for the mungbean, and 49 days after sowing every day for 6 days for the chickpea.

The extracted data from the images was the following: canopy temperature which is the maximum pixel temperature in the plant, canopy temperature depression that is the difference between air temperature, the crop water stress index of the plant in order to quantify the stress and finally the relative stomatal conductance which measures the degree of which the stomata is open and can also be used as a water status indicator.

This experiment was conducted in a green field and this methodology is more successful than conventional methods not only to distinguish genotypes and treatments but to also identify plant stress in a low-cost, reliable and non-invasive way.

A similar experiment [16] was also conducted where the plants were watered on day 1, 5 and 8 with 4 irrigation regimes, 50mL, 100mL 150mL and 200mL, the pictures were taken from various angles (front view, top view, left side view, right side view, and front angle view) at fixed locations.

To evaluate the CWSI and stomatal conductance index, dry reference leaf, wet reference leaf and temperature of canopy metrics were taken from 3 leaves from a single pot represented in a thermal image. The method to obtain the reference values was the same as mentioned before, with vaseline for the dry reference and sprinkle water for the the wet reference.

The temperature ranges only had a noticeable difference between the plants with higher irrigation(200mL and 150mL) when compared to the plants with lower irrigation (100mL and 50mL), canopy temperature of up to 3ºC on the lower registered temperature (25ºC to 28ºC) and no difference on the upper end (on the 32ºC).

Another experiment [17] controlled the temperature and humidity of the environment, keeping it at an average of 24ºC with slight variations of +-2ºC and the humidity between 73% and 84%.

The sunflower plants were also kept in small pots of 4 L Volume and 21x16.4 cm, 3 pots at full irrigation, 70% and 50%. A procedure similar to one explained before was applied, where the saturated soil was drained to achieve maximum soil capacity, and after that the soil was left in the oven for 48h at 70 degrees celcius to dry, this way we obtain the well watered weight, and dry weight.

At the end of the experiment, it was noticed not a great difference in the temperature of the plants, the plant at irrigation 50% and 75% presented around 2ºC above the plant at 100% on the 5th day of the treatment.

2.2. Modeling research

Query used: TITLE-ABS-KEY ((”Thermography” OR ”Thermal image” OR ”infrared imaging” OR ”thermal vision” OR ”heat signature” OR ”thermal infrared”) AND ”FLIR” AND (”Object detection” OR ”classification” OR ”image segmentation”) AND

("Fast R-CNN" OR "Faster R-CNN" OR "SSD" OR "EfficientDet" OR "Mask R-CNN" OR "deep learning" OR "CNN" OR "YOLO" OR "R-CNN" OR "Region Based Convolutional Neural Networks") AND NOT ("GAN")) AND PUBYEAR > 2019 AND PUBYEAR < 2026 AND (LIMIT-TO (DOCTYPE, "ar") OR LIMIT-TO (DOCTYPE, "cp")) AND (LIMIT-TO (LANGUAGE, "English"))

There are many ways to detect and classify images (or simply classify them), deep learning is a very used method of doing this, with the most popular architectures being: the Convolutional Neural Network (CNN), Region based CNN (RCNN), Fast and Faster RCNN, one-stage detectors like YOLO, Single Shot MultiBox Detector(SSD) and the RetinaNet. Architectures like YOLO are very good for real time detection and classification due to their speed, but lack accuracy when compared to R-CNNs, these are highly accurate but are slow, so each have their own advantages and disadvantages[31]. In the pre-trained CNN models we have VGG-19 for example, that was pre-trained with data from 1.2 million images from 1000 different categories [29].

The articles were filtered for it to contain only FLIR images because we used a E60 thermal camera in our project and some thermal images require specialized software to extract the thermal data, for example in [14] the thermal images were taken and were in a IRB format and to extract the thermal data in order to analyze the images, a IRBIS software was used.

2.2.1. Classification

An example of a simple classification through machine learning was the random forest(RF) and multilayer perceptron (MLP) used in the classification of whether the cotton plant was well watered or under dried down periods as it was mentioned before[14].

The dataset had 648 images, 2 images per leaf from 3 plants of the same genotype with 27 different genotypes, twice and on 2 treatment days.The dataset was thinned down to 419 images, with balanced classes (well watered, dried down). The thermal data was processed using excel and R 4.3.1 with dplyr and tidyverse packages and for the graphics ggplot2 and ggrepel was used. On the excel each pixel had 2 representations, the RBG color and the thermal value. To extract the leaf from the image, the authors used a fast Fourier transform with a binary mask to keep all the wanted pixels whose value, after that the inverse transform was applied in order to compute the leaf mask. For the classification the authors used a multi-layered perceptron and obtained a overall accuracy of 78%

A similar mask method was also used in the experiment [15], where the infrared image was segmented to 0 and 1s (0 being the zone where there is no plant, and 1 being the zone of the image where the plant is displayed) and then merged with the real infrared image, this way we are able to get only the temperature of the plant with no background noise.

Besides machine learning we can also use deep learning CNNs which are typically built with three main types of layers: Convolutional and pooling layers work together to extract features from the input, while fully connected layers map these features to the final output.

This process is applied to thermal images as follows:

- **Convolutional Layers:** These layers use small matrices called kernels or filters to scan the thermal images. Each kernel is trained to detect specific features such as temperature variations, edges, or textures within the image. By performing the convolution operation, these layers produce feature maps that highlight areas of interest.
- **Pooling Layers:** These layers reduce the size of the feature maps through max pooling, which selects the maximum value within small regions. This down-sampling reduces computational load and helps the network focus on the most prominent features, making it robust against minor image variations.
- **Fully Connected Layers:** Once the feature extraction is complete, the network flattens these features into a one-dimensional array. The fully connected layers then process this data to determine the likelihood of the image belonging to one of the existing classes. The output is a classification prediction.

Besides these 3 main layers there can also exist a dropout layer which prevents overfitting, dropout randomly ignores a proportion of neurons during training, forcing the network to learn more robust features, the activation function that after convolution, activation functions like ReLU (Rectified Linear Unit) are used to introduce non-linearity, enabling the network to learn complex patterns.

A kernel can be referenced as a small matrix of weights used in the convolutional operation and it is also called a filter, these filter slide over the image and perform multiplications and sums to produce the feature map, each filter can have different weights that will determine which features are extracted and how, allowing the network to learn and recognize patterns in other images.

The pooling layers decrease the size of the feature map, learned parameters and performed computations in order to control overfitting and allowing the network to extract only the dominant features as it was previously mentioned. The outputted feature map is the final convolution or pooling layer, and these are usually flattened into a one dimensional array, fully connected into a dense layer so that every input is linked to every output. The total output nodes of the last fully connected layer is the number of classes, for example 2 classes means we have a binary classification [18].

Usually input images have to be resized to fit the input layer for example in this neural network, the image was resized to 320x240 pixels. The authors trained a custom CNN with one layer of 32 3x3 filters followed by a 2x2 max pooling, then two more layers with the same structure but 64 filters. The last layer(flatten) was then followed by 4 dense layers of 64,128,256 and 2 neurons (the last one is for the classification output) with a

dropout to avoid overfitting(excessive training) on Google collab, which allows the usage of their GPU(Graphics Processing Unit) for faster training. They had a total of 286 images of two classes and fairly balanced (206 to 180), and split it into 80% train 20% test, ending with an accuracy of 100% on the train and 82% on the test [18].

A transfer learning example, which is a pre-trained model in order to leverage the resources needed to train the new model, an example of this method was in the article [19] where the authors used 4 transfer learning models, the VGG-16, Efficient-Net, ResNet and Inception-V3, on a training dataset of 2704 audio files and 676 validation files for for "coughing" detection, this was a 80:20 split, the authors ensured a balanced dataset with 1500 positive and negative files for cough classification, the authors created 2 distinct CNNs for each task, after the cough detection model gave positive feedback then the file would go to the cough resemblance model for further classification.

The files were labeled as positive or negative for COVID-19 accordingly and the audio files were then pre-processed generating Mel-spectograms in RGB format from the audio, resizing them to 224x224 to fit the transfer learning input layer and converted to NumpPy array containing the features and the labels to use as input for the CNN.

This approach for transfer learning was the following: get the pre-trained model (like the vgg16), create the base model, freeze its layer, add new trainable layers, train these new layers and improve using fine-tuning, it was also used dropouts to aid in regularization, it sets a fraction of inputs to zero in order to mitigate overfitting and create more robust and generalized representations as it was mentioned before.

This was a binary problem on both models and both cough detection and cough classification all the transfer learning models performed well, predominantly the VGG16 and ResNet [19].

Sometimes we are not able to obtain more images, in that case there is a process called data augmentation that helps increasing the size of the dataset through the creation of variations of existing images using rotations, horizontal and vertical flips, zoom in and out, changing colors.

For example using 3 different degrees the authors in this paper [20] were able to go from 500 images to 1500, they then proceeded to test with machine learning algorithms like the random forest, kNN and SVMs with some feature extractions but also deep learning algorithms like the ResNet50, DenseNet121 and a custom CNN model. For the custom CNN the authors used a learning rate of 0.001 and a batch size of 64, it was used a 80/20 train/validation split, in the results their custom model was the best performing.

Another example of a CNN model was in the project [21], where the authors used a more recent architecture, the VGG-19 which was trained in the ImageNet dataset, this model had 19 layers consisted of fully connected layers, max-pooling and convolutional

layers, more convolutional layers were added after for fine tuning. Each convolutional layer had a filter size of 3x3 and was followed by a RELU (rectified linear unit) activation function and a max-pooling with a filter size of 2x2. This dataset was created by a FLIR thermal camera with 500 images from 10 classes. These were resized to 224x224 which is the input size required by the VGG-19. Once again the CNN method proved very effective with an accuracy of almost 100% and of simple implementation.

The next proposed model [22], was a pig treatment classification. The dataset consisted of 62800 thermal images with 4 labels, isolation after feeding (IAF), isolation before feeding (IBF), paired after feeding (PAF) and paired before feeding (IBF).

These images were resized to 112x112 and the model trained for 100 epochs with a learning rate of 10^{-3} and a batch size of 32. The dataset was split into 60, 20, 20 for train, test and validation. The authors trained a LeNet5, AlexNet, VGGNet as well, to compare with their custom made model using convolutional layers followed by Leaky ReLU, a dropout layer was also applied to prevent overfitting, and softmax as the activation function in the last layer. This model had 5 layers with learnable parameters and every filter had a size of 3x3 an. The proposed model behaved really well, only being rivalled by LeNet5, both having almost 100% accuracy, F1 score, precision and recall.

This experiment[23] was conducted to classify stress using thermal images with a smartphone. For that they used a mobileNetV2 which is a lightweight deep neural network, with a batch size of 32 and for 100 epochs. The model was split into train, validation and test and used three callbacks to monitor training: model checkpoint,early stopping, tensorboard callback. The mobileNetV2 is the second version of the mobileNet and has less amount of parameters than the original one, making it much faster due to reduced size and complexity. They had an average accuracy of 91% with a similar validation accuracy, meaning the model did not overtrain. In this paper the mobileNetV2 is compared to the VGG16 and said to show superior results while being light and able to run without GPU support, the VGG16 also presents good results but has a much higher amount of parameters making it harder to run on machines with low hardware.

Another approach with some changes was [24], used rotations and flips to increase the dataset by 1600% and some Gaussian noise to make it more resilient to motion blur, these changes do not alter the pixel intensity values keeping the thermal image intact. The images were resized to 320x320 and categorized as healthy or faulty blades (the dataset contained 500 images on each category).

The dataset was then split into train, validation and testing (80% of the total dataset for training , 10% of the training dataset for validation and then 20% of the total once again for testing) and changed the learning rate between 1×10^{-5} and $1\times10^{-4}$for stable training, batch size between 16 and 128 to optimize computational resources and gradient

estimation,weight decay within 1×10^{-4} to 1×10^{-2} to avert overfitting and set a early stopping to monitor validation loss and stop training after no improvements over a set number of epochs.

Various transfer learning models were trained with the help of a vision transformer model to reduce the training time. The performance across the board on all models was very good, but the most notable approach was the ensemble model, this combines the other models having them working together to solve a common problem, compensating each others weaknesses creating a more robust and accurate model, this model used different weights of Vision Transformer and DenseNet161 with ratios of 60 to 40 and 70 to 30.

The approach on [25], the authors applied a CNN architecture on two different datasets: the FLIR dataset and Seek Thermal, both had 3 balanced classes (cat, man and car) with seek thermal being the larger dataset with 6414 and FLIR with 1014. The models used were the DenseNet, MobileNet, and YOLOv4 for comparison and the Inception v3 for classification.

The inception v3 had some upgrades from the v3, batch normalization, which normalizes all feature maps to increase stability and reduce overfitting, factorization to reduce the number of parameters by splitting larger convolutions into smaller ones and grid size reduction which uses both pooling and convolutions each with a stride of 2 reducing the grid size. For the split the authors used a 80/20 training/validation split.

On the evaluation of the classification results, on Seek Thermal dataset, both the mobileNet and inceptionv3 had perfect scores classifying correctly all images. On the FLIR dataset, inception was still the best with 99% accuracy, precision and recall but the densetNet also had good results, with 93.3% accuracy and a 90% precision and recall.

A custom CNN was also trained, on the Seek Thermal dataset it behaved quite well, with 96% accuracy, recall and precision but on the FLIR dataset it had those values under 75%.

The authors on article [26] performed only a custom CNN, they used a real time video and treated each frame as a different classification due to the single image input properties of the CNN. It was a 3 layer CNN with 64 3x3 filters for the first and 128 3x3 filters for the second, then there is a flatten and 50% dropout a dense layer with 256 neurons all the layers use a relu activation function, ending with a sigmoid (its good for binary classifications) final output. For the training the authors used a 5-fold cross validation with a learning rate of 0.01 for 32 epochs. 4 different models were trained 4 always with a binary classification and some data augmentation with Keras ImageDataGenerator due to the small size of the dataset taking advantage of rotations, zoom, shear width shift and height shift.

One model to classify if the person has normal or reduced mobility (91% accuracy on average on 12 frame), another to check if there is 1 person or more than 1 (85% accuracy

on average on 12 frame), a model to classify the gender (77% accuracy on average on 12 frame) and another to check if the person is an adult or an elder(75% accuracy on average on 12 frames).

In this example the authors try to create a classification backbone for the YOLOv9 but in order to improve the details on the image they applied constrast-limited adaptive histogram equalization (CLAHE) or image contrast enhancement (ICE). When compared to the normal image, there is a huge difference in the visualization and this allowed the model to detect the object more easily.

The original YOLOv9 object detection model uses the Darknet-53 as the backbone of the model but the authors built a CNN with an input channel of 32x32 compared to the most common in transfer learning 224x224x3 to better fit their device resolution.

It possessed an adam optimizer learning rate of 0.001 with batches of 64 images during 80 epochs for training and 20 for retraining. When comparing the quality of the model when trained under normal raw images or images with normalization or upgraded by CLAHE or ICE we can see a decent improvement in the accuracy of the model.

On the raw data it has 87.9% accuracy, on CLAHE and ICE it has 92.3% and 93.5% accuracy respectively. With transfer learning on the obtained dataset it improved a little more, ICE went from 90.7% on that test to 96.4%, same happened with CLAHE that also improved from 90.4% to 93.1%, but the best improvement was from the raw dataset where it had 40.8% accuracy it improved to 90.5% when using transfer learning [27].

2.2.2. Object detection

Object detection uses algorithms like the Yolo or Faster R-CNN in order to locate the objects, determining their size and position in the image and classify them, after that we resize the images and split it into training and validation sets [28].

In order to train a quality model, we need to draw bounding boxes on the objects we want to detect and classify in our training images and label them.

To evaluate the quality of the trained model we assess the Intersection over Union or IoU for short by comparing the predicted bounding boxes with the ground-truth. If the IoU score between the predicted bounding box and the ground-truth is higher than the chosen threshold(in this case 0.65) it is considered a true positive.

To evaluate the overall performance of the model, we calculate the mean average precision(mAP), which is the average of the precision values(AP) across all object classes and various IoU thresholds. For classification it is used a confidence score(in this case of 0.25) to classify the bounding box. Besides average precision and mAP there is also model speed, precision and recall to take into account[28].

These models are composed by a input layer, a backbone which is the deep neural network composed of mainly convolutional layers, whose main objective is to extract the essential features, a neck which sits between the backbone and the head, whose job is

to combine feature maps from different stages of the backbone and the head which is responsible for making the actual predictions like bounding box coordinates and class probabilities[29].

This article used the YOLOv7, this model, uses 40% less parameters in the hidden layer than YOLOv4, making it faster but it also manages to achieve a higher precision than the previous model.

The YOLOv7 usually uses VGG-16, ResNet-50 or RetinaNet as a backbone but the authors used VoVnET and ELAN for the backbone. For the dataset, it was used UAV images with a resolution of 640x512 which was later resized to 640x640, it contained 3807 annotated images of people, with 2358 positive and 1449 negative images. The images were annotated and then it was performed data augmentation (flip, crop, 90° rotation) to obtain 9369 total images. This was all done with the help of Roboflow.

The training was performed on Google Colab. It was done with a batch size of 16, initial learning rate of 0.01 with weight decay and momentum set to 0.937 and 0.0005 for 40 iterations. The images were split into train, test and validation and achieved mAP of 72.50%, with a precision and recall of 81.7% and 64%.

When compared with other models, the authors say the performance of their model was influenced by the type of data used, the best model was trained with nitid images at a closer distance when compared to an image from their dataset in which the authors model was trained. Another thing to keep in mind is that their model was a lot faster, having 160 frames per second compared to the top model (20 frames per second). But still the model performs quite well, when we take the object size into account[29].

In the project [30] the authors test various versions of the YOLOv3, YOLOv4 and YOLOv5, overall all these architectures have good results. These can all be preprocessed in similar ways, by simply annotating them and then resizing the images (usually to 640x640). After training these models we proceed to evaluate them by measuring their speed, mAP, precision, average precision and recall.

YOLOv3 is an example of a detection framework, it consists of two parts, a backbone for feature extraction and a prediction network to fuse the features and generate the detections and their locations, in this paper the backbone used was the SqueezeNet which is faster and simpler yet it achieves the same level of detection performance among its peers with much less parameters.

The models the authors trained had an initial learning rate of 0.001 and further decreasing to $1.0e-5$, it was split into 80/20 training test, they were trained on a FLIR dataset, a local dataset and a combination of both. The model1 did not present very good results on FLIR dataset but was good on the local data one, with an average precision of over .9 on all 3 classes (car, person and dog), the model2 did not present good results, both model 1 and 2 were trained each on a single dataset which is why their results were

not quite as good, when tested on the dataset they were trained on the results were good, which is the example of model2 on the local data but when it comes to generalize they present poor results. Model3 presented good results on both datasets since it was trained on both data combined, with an average precision of 0.8 across classes on the flir data and 0.98 on the local data [30].

Another method of object detection is to use two architectures like the YOLOv8 and EfficientDet which are more modern models. YOLOv8 was the latest YOLO version at the time and the best, the authors used a custom version of the CSPDarknet53 architecture with 53 convolutional layers and as it was mentioned before, they paired the YOLOv8 with EfficientDet.

Before training they applied a variety of data augmentation like rotation, scaling and flipping to increase the diversity of the dataset and then independently fine tuned each, then they overlap, combine the bounding boxes of each and then refine the detection results using a non-maximum suppression algorithm, then they keep the bounding box with the maximum confidence score (the other values are discarded).

The results obtained were very good with the proposed ensemble achieving better results than each model independently, but not by a very notable amount (the most notable increase was the comparison of the mAP from the efficientDet (0.874) to the mAP of the ensemble (0.931) on the RGB dataset, the rest of the stats across both RGB and thermal dataset increased between 1% to 4%). On the other hand the ensemble model is much slower than both the independent models (32ms ensemble, YOLO with 10ms and efficientDet with 20ms) [31].

In this paper[32] the authors pointed out one of the weaknesses of models like the YOLO or SSD networks, is that they have some difficulty detecting smaller objects, this was also mentioned in paper[29] where one of the datasets presented better results due to the objects in the image being closer reinforcing the idea that detectors like YOLO benefit from larger-scale objects.

The authors proposed a faster R-CNN architecture with a feature pyramid networks. The region proposal network will suggest regions where the there might be an object while the head refines the bounding box and classify the object. The feature pyramid networks will improve the detection of smaller objects by combining feature maps from different levels of the feature extractor. The backbone used was the ResNet-50, and they use a fusion method that concatenates the RGB and the thermal feature maps. With a split of 8800 training images and 1200 test images, with 60% daylight images and 40% night images they were trying to detect persons, cars and bicycles. While the conjunction of the RGB images with the thermal images did not upgrade the model very much, in the flir

thermal dataset it actually decreased the mAP@50 of the model, the feature pyramid network improved the overall mAP@50 of the model by up to 17.5% on the FLIR dataset[32].

To address blurred edges, monochromatic color and indistinct object-background differentiation the authors proposed the usage of CLAHE, this technique as it was mentioned before in classification is used to enhance the image reducing the noise and alloying the model to better classify the image, in this case it will help it detect the object within the image.

Since it may also increase the image noise it was applied a bilateral filtering as well which removes the noise without blurring the edges, after that the image brightness and contrast was improved, increasing the pixel value of the grayscales with higher value and decreasing the ones with lower pixel value.

An automatic gamma correction was then applied to enhance the dark details of the image, if the gamma value is under 1 the image gets brighter making the dark parts clearer, if the gamma is over 1 the image gets darker and may reveal bright details[33].

This is a multimodal approach [34] that uses both thermal image and RGB, through early fusion or late fusion.

The early fusion combines the RGB image and thermal image into one multi channel image, the late fusion trains 2 models on the different image types and then combine their detection results, if both detectors contain a bounding box that match the same ground truth only the thermal detection is kept.

The results were good, but simply using the thermal image in the baseline model demonstrated near the same results, having mAP@50:95 results slightly better than early fusion and slightly worse than late fusion on the FLIR dataset.

Another example is the M-YOLO, with a lightweight mobileNetV2 for the backbone which uses only 17 layers and has a much faster processing while maintaining good performance.

The authors considered global average pooling (GAP) a good baseline model so they applied a global context information aggregation model which consists of two main modules, first they apply a global average pooling in three sub-regions to capture both local details and global context, preserving more spatial information than a single GAP and then they apply a split attention module to suppress secondary information highlighting the correct feature regions in the process.

Once again as it was mentioned before YOLO has a problem with small objects as such they usually adopt the feature pyramid network as the output network but this will cause some information loss on infrared images, the different layers will have different types of information and direct fusion will reduce the quality of multi-scale representation. To

overcome this they combine features from shallow layers with deep layers. As a result it improved the accuracy of the model by 5%[35].

This project [36] was the comparison of 2 simple YOLO models, the YOLOv5s and YOLOv7. The dataset was composed with 9958 images and split into train and validation (8820 to 1138).

The annotation files were converted from json to YOLOv5s and YOLOv7 pyTorch format, and the images were resized to 640x640 to match the YOLOv7 and 416x416 for YOLOv5s, the models were trained after.

The YOLOv7 obtained quite good results with a 82.4% accuracy, with its lowest accuracy being the prediction and detection of bicycles (66.8%). YOLOv5s obtained 71.8% accuracy, it being around 10% lower than the more modern model YOLOv7, but we have to keep in mind that YOLOv5s is also a smaller and faster model.

Another similar method to object detection is image segmentation.

The authors used a U-net, the architecture has 2 paths, the encoder which is used to extract the spatial features from the image (a group of convolutional and max pooling layers) and the second path which is used to create the segmentation map, colouring each object in the image. Usually the U-net uses 3 channels (RGB) but in this case it has 4, the extra being a thermal one, so the first and second channel are the thermal image and the third is a gray scale version of the colour image. It will essentially label which pixel belongs to which class[35].

CHAPTER 3

Experimental protocol

3.1. Plants and water stress

For the plant to sustain its activities, it must perform 2 key processes: photosynthesis and transpiration.

In photosynthesis, the plant converts light into chemical energy that will later be used to fuel the plants activities. This reaction then synthesizes carbohydrates from carbon dioxide and water, releasing oxygen (O2) in the process.

Transpiration is crucial for regulating temperature dissipating heat through the release of water vapor from microscopic pores called stomata. These pores also enable the plant to exchange moisture and gas with the atmosphere permitting the diffusion of CO2 into the plant for photosynthesis [1].

This process creates an evaporative cooling effect that reduces the overall temperature of the plant. This process will also cause the alteration of the osmotic pressure within its cells which facilitates the upward movement of water and mineral nutrients from the roots to the rest of the plant, supporting both its metabolic processes and structural integrity [1].

Both transpiration and photosynthesis can be roughly observed on figure 3.1.

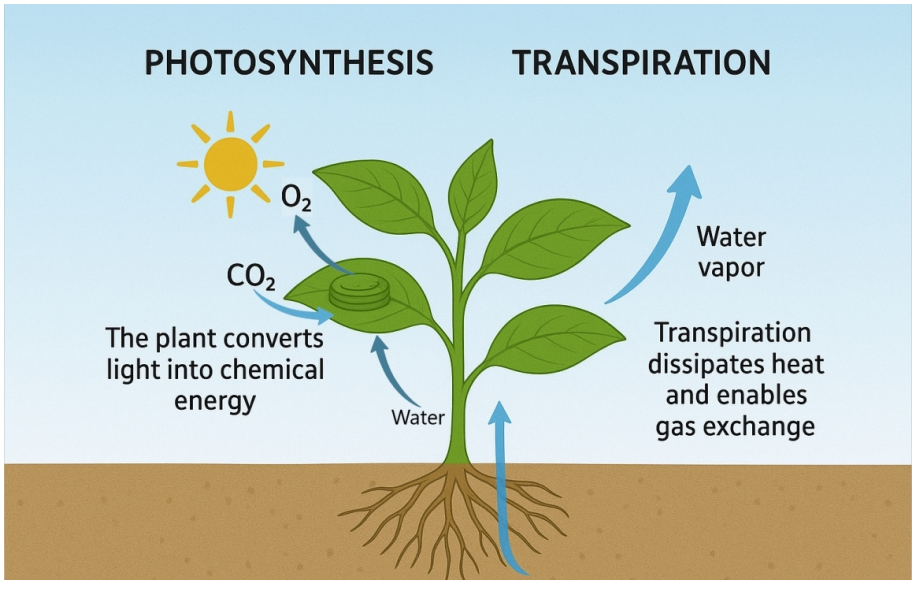


FIGURE 3.1. Photosynthesis and Transpiration

When under water stress the plant will start to close these pores, which will not only reduce transpiration but it will also limit the assimilation of CO2, this will reduce the plants ability to do photosynthesis and consequently inhibits its biosynthesis [6].
Water stress progresses through several stages:

- **Mild Stress:** The plant starts closing its stomata to reduce water loss.
- **Moderate Stress:** Stomatal closure increases, significantly impairing photosynthesis.
- **Severe Stress:** This can cause permanent damage to the plant's photosynthetic machinery.

The longer the drought period, the harsher the consequences it will sustain besides damaging its photosynthetic capacity in the agricultural field this will also cause lower production yields, reduced crop quality or the death of the plant [1].

3.1.1. Irrigation Experiments

For this project the images were acquired using a FLIR E60 handheld thermal camera (figure 3.2).

This is a professional-grade, infrared camera it has many applications one of them being the agricultural domain due to its high sensitivity and portability but it can also be applied in other fields such as the medical field as it was previously mentioned. Here are the specifications of the camera

FIGURE 3.2. FLIR E60 handheld thermal camera

- Thermal Resolution: 320 × 240 pixels
- Spectral Range: 7.5–13 μm (long-wave infrared)
- Thermal Sensitivity (NETD): $<$ 0.05°C at 30°C (50 mK), enabling detection of subtle temperature differences
- Temperature Range: -20°C to 650°C
- Accuracy: ±2°C or ±2% of reading
- Field of View (FOV): 25° × 19°
- Image Frequency: 60 Hz
- Focus: Manual focus for precise image clarity
- Image Storage: JPEG format with embedded temperature data (radiometric images)
- Display: 3.5" touchscreen LCD with visual and thermal image fusion

All objects emit infrared radiation as a function of their temperature. The FLIR E60 uses a microbolometer sensor that detects this infrared energy in the long-wave infrared (LWIR) range—typically between 7.5 and 13 microns. The detected infrared radiation is then converted into electric signals, which are then processed to create a thermal image. Each pixel in the image represents a temperature reading, allowing the camera to display heat variations across the surface of the object.

The E60 is able to display in the image observed the minimum and maximum temperature in the image or within a custom temperature box as well as the temperature on desired spots. On figure 3.3, we have an example of a custom box displaying both the maximum temperature within it and on figure 3.4 we see and example of a marked spot (in the center of the image) which will show the temperature on that specific marker in the image, this marker can be moved to different places in the images and we can add more custom markers if needed.

On figure 3.5 we have an example of an image with a custom box measuring two different point markers, one displaying the maximum temperature and another displaying the minimum temperature within the box (the different display from the image in figure 3.5 is different from figure 3.3 and 3.4 because this figure was taken as a snapshot from the camera directly into the phone using the flir tools app and the other two images were taken from the camera into a ssd card but both methods possess all mentioned functionalities).

Keep in mind, the thermal data is only saved if the image is taken into a ssd memory card, for we tried using the FLIR TOOLS mobile app and it only saved a snapshot of the image with no thermal data within it (if the image is saved into the ssd memory card and uploaded into the FLIR TOOLS app in the desktop it will still retain the thermal data).

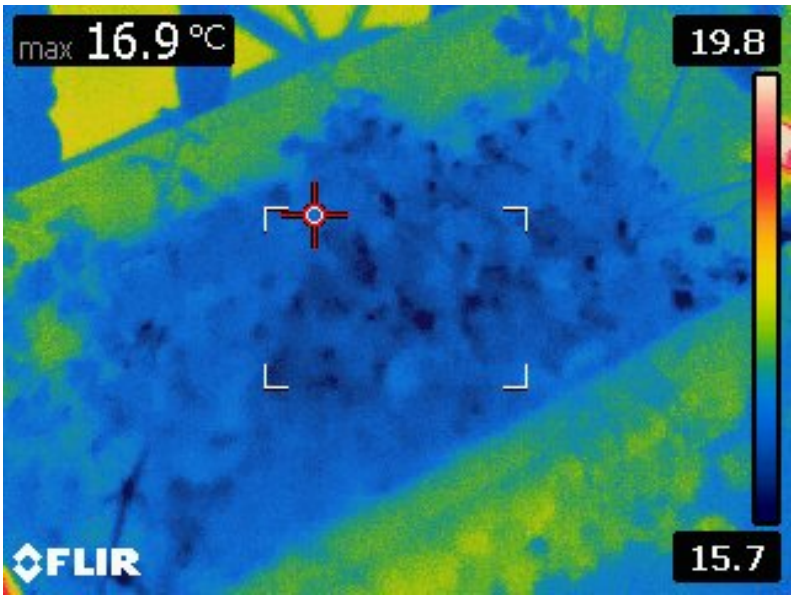


FIGURE 3.3. Thermal image with temperature box displaying maximum temperature within it

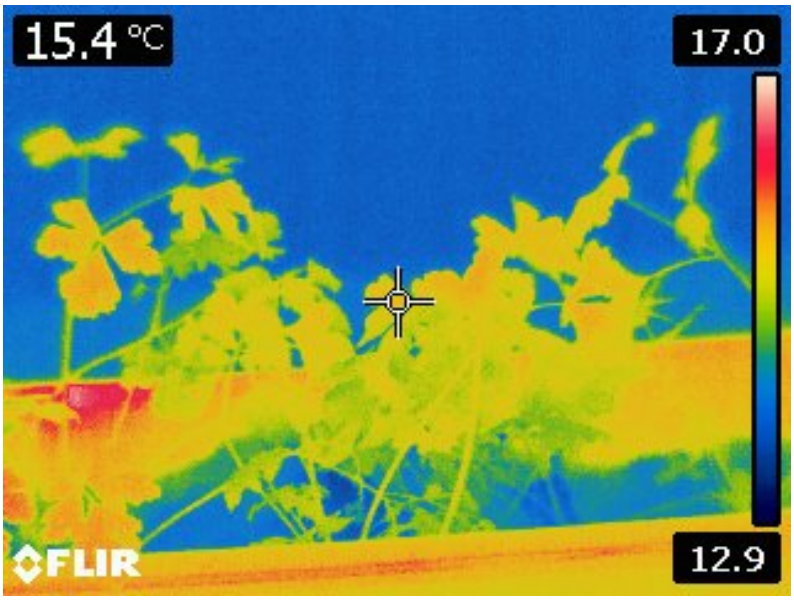


FIGURE 3.4. Thermal image with center marker

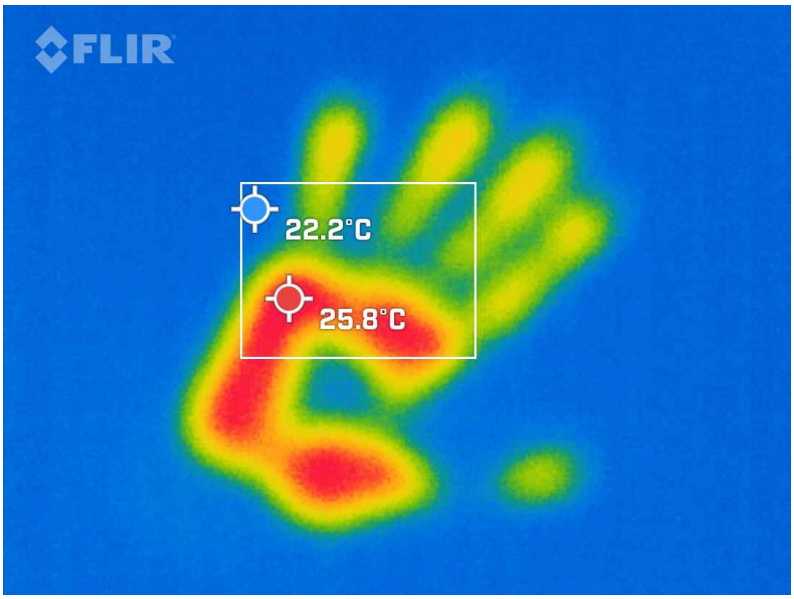


FIGURE 3.5. Thermal image with double marker displaying minimum and maximum temperature on a custom box

To take the pictures it is required to define the emissivity of the image, this measures how effectively a surface emits thermal radiation compared to a perfect emitter, known as a black body. The value typically falls between 0.01 and 0.99, depending on the composition of the material and surface finish. A perfect black body has an emissivity of 1.0 while a highly reflective surface can haven an emissivity as low as 0.1 [10].

According to article [3] the emissivity of most plant surfaces range between 0.97 and 0.99 so our images were taken using an emissivity value of 0.98.

The common parsley (Petroselinum crispum) was selected for this study, serving as the foundation to our dataset. It was chosen due to it being a n agricultural plant grown for food commerce purposes, easy access and suitable green foliage which facilitate the clear observation of the effects of water stress on the thermal image.

For the first experiment we used 6 pots each with a plant under 50 ml of water. The pots were grouped in pairs, one pair was watered once a week, the second twice and the third pair was watered 3 times a week.

This experiment occurred during winter but because of the cooler weather, small surface size of the leaves, it was hard to observe any difference in the measured temperature even though some of the plants were withering due to lack of water. This experiment reinforces the idea that cold humid weather has a direct impact in the temperature of the leaves.

With the new insights later on in 27 February 2025 we got a precision scale to measure the soil, 3 bigger pots (each pot weighs 290 grams). For this experiment we transferred

the parsley plants to the new pots with 2kg of substrate and replaced the old soil. To obtain the maximum soil capacity we measured a cup with substrate, being its maximum capacity 70 grams, that soil was then saturated with water and few incisions were performed on the cup to drain the excess water, achieving 120g which is the full 100% soil capacity.

The 70 grams of soil were then converted to 2000 grams as it was the amount used on the bigger pots and irrigated according to the conversions, giving us the following values:

TABLE 3.1. Soil weights at different water capacity levels

	Soil weight
100% water capacity	3768.6g
50% water capacity	3054.3g
25% water capacity	2697.1g

The pictures obtained a decent difference between the plant with 100% moisture compared to the rest of the pots, but there was little difference between the pot with 25% moisture and the one 50% moisture.

A fertilizer was also mixed into the water (Compensal 12-4-6) by 1 liter of water for each 2 milliliters of fertilizer after the first experiment to allow the plants to grow more and faster since the lack of leaves was one of the set backs in the previous experiment.

The fertilizer was composed by:

- nitrogen(N) 12%,
- phosphorus pentoxide(P2O5) 4%,
- Water-soluble potassium oxide(K2O) 6%,
- Water-soluble magnesium oxide(MgO) 0.2%,
- Boron (B) soluble in water as boric acid 0.02%,
- Water-soluble copper (Cu) chelated by EDTA0.01%,
- Water-soluble iron (Fe) chelated by EDTA 0.02%,
- Water-soluble manganese (Mn) chelated by EDTA 0.01%,
- Molybdenum (Mo) soluble in water as sodium molybdate 0.005%
- Water-soluble zinc (Zn) chelated by EDTA 0.005%

One limitation that was observed in this method was the fact that the soil that was used (was taken directly from a soil bag) already possessed some moisture, inflating the water percentage on each pot.

In order to remove this humidity and perform an accurate conversion we would require a oven to dry this soil for 72h at 105 degrees as it was mentioned in the article [15],

therefore using a moisture sensor would the more feasible and accurate method.

We can see on figure 3.6 examples of pictures from this experiment on 27th February, during cold temperatures (15 degrees celsius), rainy weather and a higher percentage moisture than what was expected due to the existence of previous moisture on the soil.

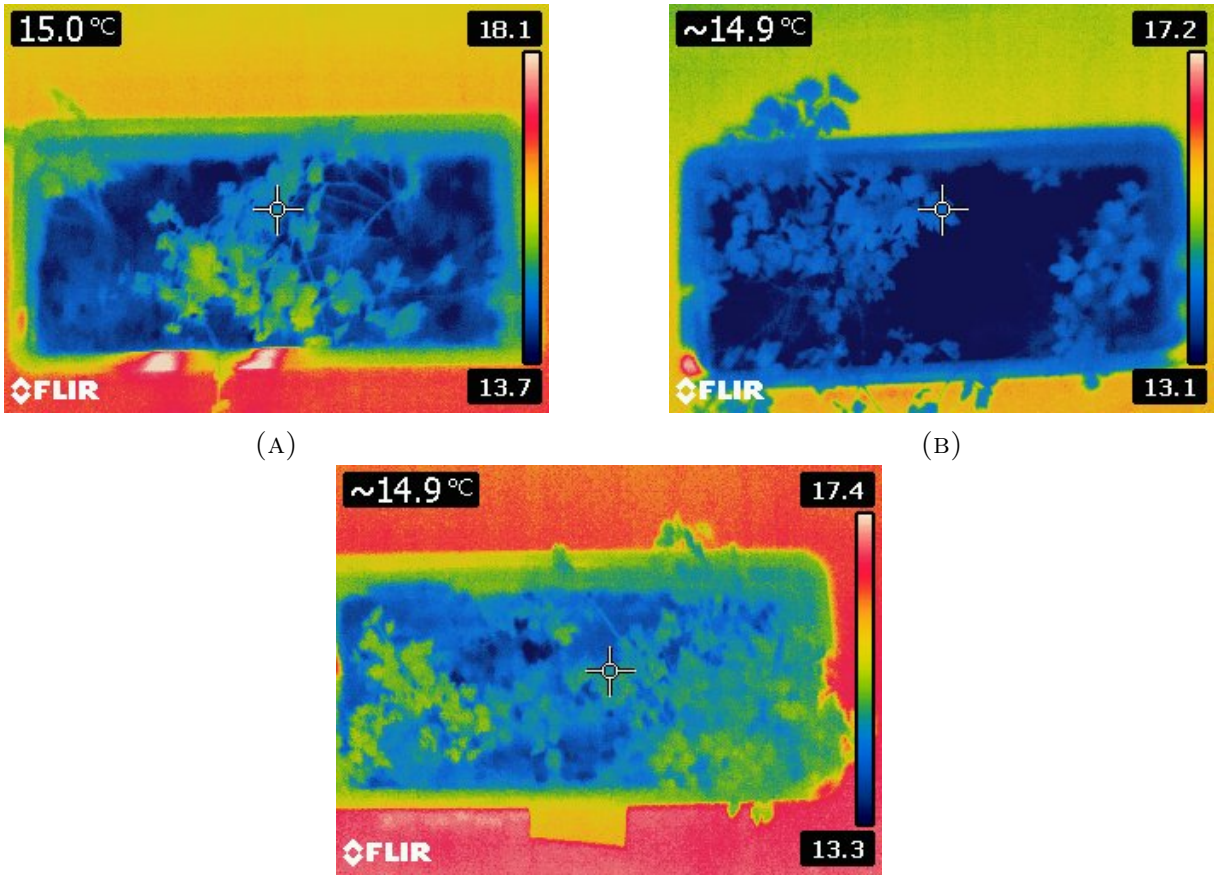


FIGURE 3.6. Comparison of plant stress levels cold weather, with more moisture than expected. a) Plant with lowest irrigation; b) plant with medium irrigation; c) Plant with highest irrigation

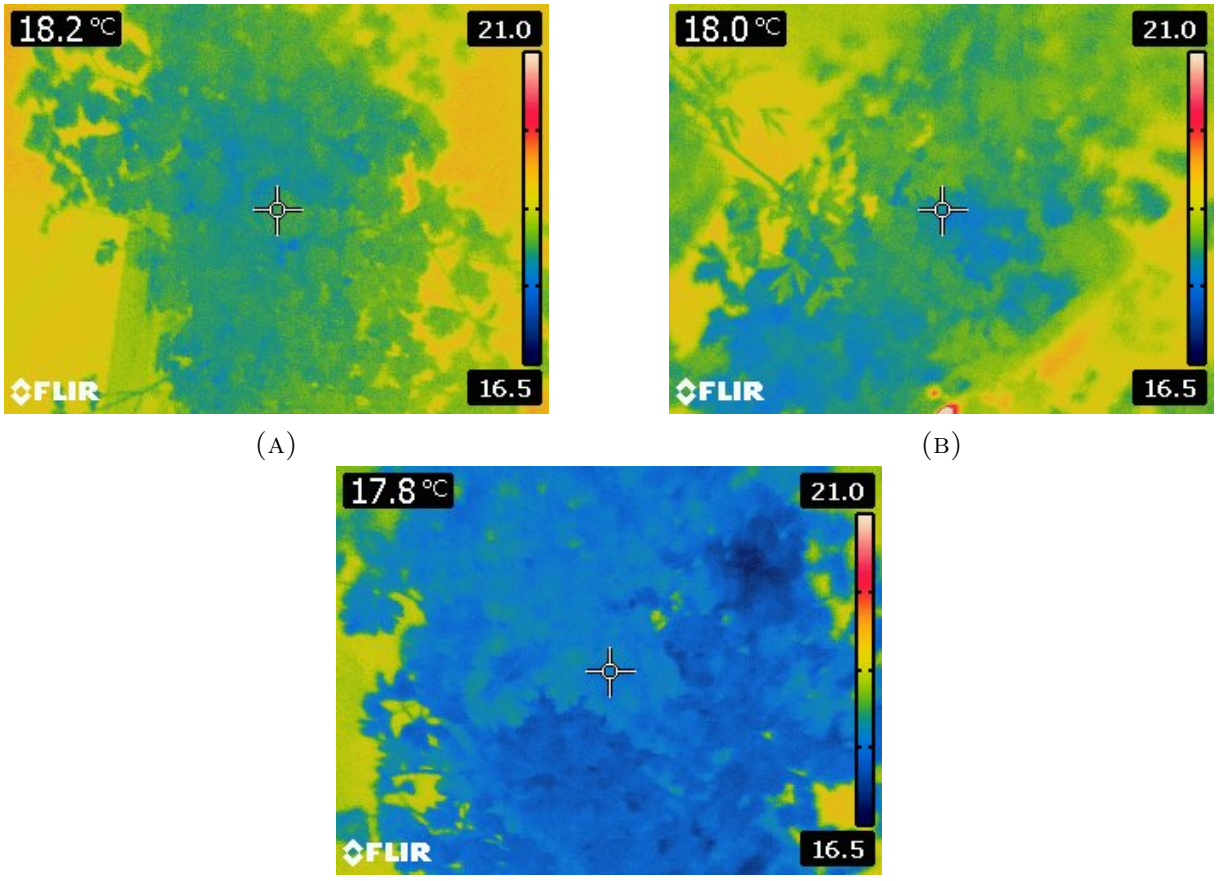
A new experiment was conducted on 22 April with a moisture sensor, the moisture level in the pot at 100% was lowered to 85% of the sensor value to prevent root rot, since the plants in the previous experiment at 100% soil moisture were quite flaccid due to excessive water.

The values were then adjusted using the soil moisture sensor to achieve more accurate results and a bigger difference in the temperature of the leaves between plants. After the adjustments the pot with higher irrigation weighted 3300g to 3400g, the pot with medium irrigation 2200g to 2300g and the pot at low irrigation 1750g to 1850g as it can be observed in the following table.

TABLE 3.2. Final soil weights at different water capacity levels adjusted with the moisture sensor

	Soil weight
85% water capacity	3300g to 3400g
50% water capacity	2200g to 2300g
25% water capacity	1750g to 1850g

During this new experiment we still had some difficulties observing a difference during rainy humid day as it can be demonstrated in figure 3.7.



(A) (B)

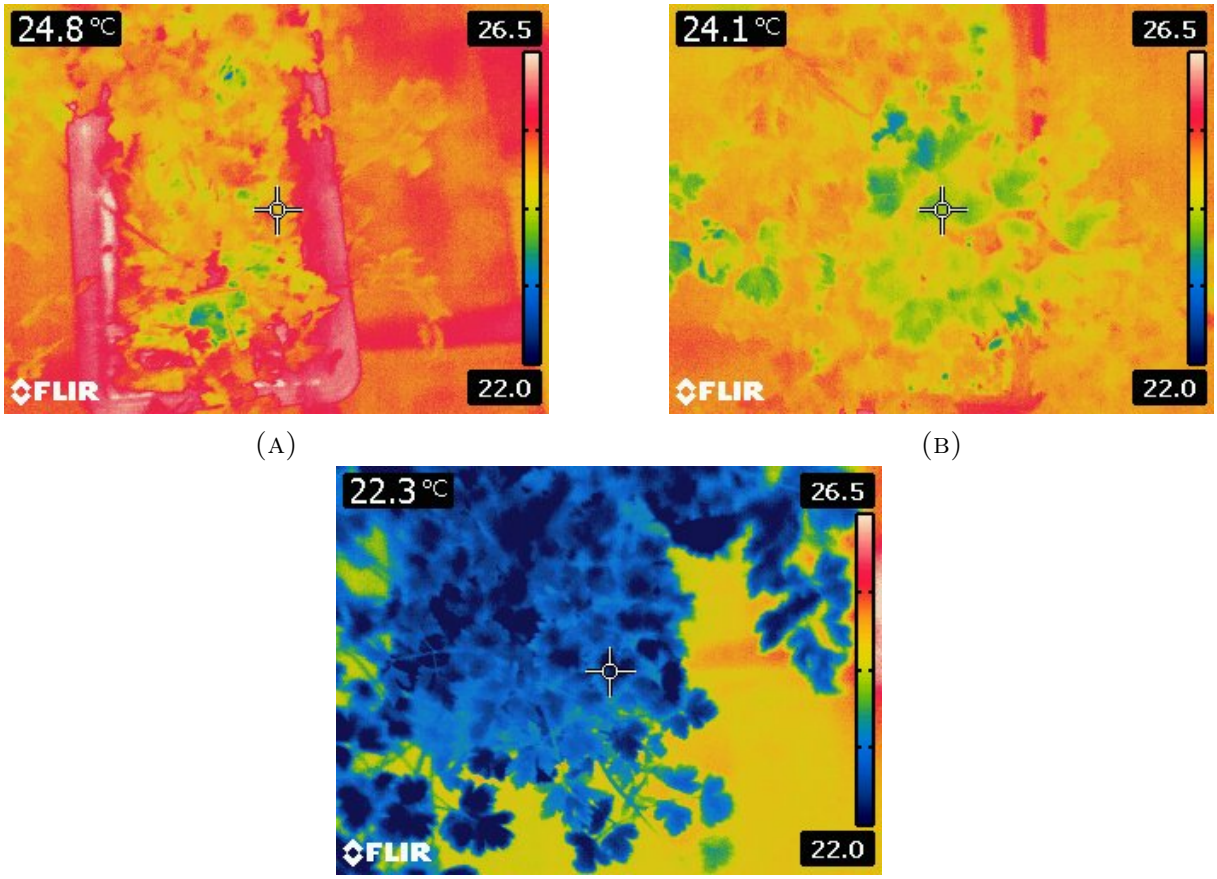
(C)

FIGURE 3.7. Comparison of plant stress levels under rainy humid weather conditions a) High stress plant; b) Medium stress plant; c) No stress plant

On figure 3.8 there is a very significant visual difference between the low stress and high stress images. On the other hand the plants at medium stress level, as expected, are on the middle of the temperature spectrum.

These images were taken during a sunny and dry weather with higher temperatures, during this weather the plants with low and medium irrigation would be quite flaccid due to the need of water while the plant with high irrigation would still be turgid.

On the thermal images, it can be observed that the plants at high stress have a higher temperature than room temperature, the leaves of the plants at medium stress level tend to be at room temperature and finally the plants with no water stress are below room temperature (during our experiment on dry, sunny weather conditions they tend to be around 2 degrees below room temperature).



(A) (B)

(C)

FIGURE 3.8. Comparison of stress levels under sunny and dry weather a) High stress plant; b) Medium stress plant; c) No stress plant

To check the water stress on each plant, we performed the difference between canopy temperature and room temperature. This information in conjunction with the field capacity of each pot allowed us to estimate how stressed the plant is.

Since the used method for checking the water stress the plant is experiencing is not optimal, for future projects, either the crop water stress index should be taken into account since its the most used metric or checking the plants stomatal conductance according to paper [3].

For the irrigation method and periods, the plants were weighted and had their moisture percentages measured across 9 different spots on the pot to obtain information on the current global moisture percentage on each of them.

The plants were then slowly irrigated with a syringe so with some waiting time in between to allow the water to infiltrate the soil. This way the water lost to evapotranspiration is refilled keeping each pot under the specified soil moisture levels. The plants were irrigated at 12:00 and 19:15 every day to allow consistency in the water stress conditions.

3.1.2. Final Conditions for the Photo Session

For our photo sessions, we tried various methods, settling with the following to achieve the best and most consistent results:

The session started at around 18:30 just before the irrigation period, this way the thermal values are not impacted by it. Before taking any pictures the plants would be put under shade for a small amount of time, this way there would not be bias towards shaded leaves and unshaded and their temperature would stabilize with the room temperature (this approach would stabilize the canopy temperature across the plant and water stress variation was still observable from pot to pot).

During the photo session, there was a designated spot where the pot would be in order to keep the conditions even for each plant and 100 photos would be taken, 40 photos from the front of the pot, and then 30 from each side, all these photos were taken the same way across all pots in order to achieve a balanced dataset and allowing us to capture all parts of the plant. All of them taken from an above point of view and under shade this way the temperature variation from solar radiation is also avoided.

The room temperature would also be measured and saved in order to compare it with the leaf temperature and later use it to normalize the daily temperatures. For the final dataset, we were left with around 500 images per class to create the models.

CHAPTER 4

Data preparation, Modeling and Evaluation

4.1. Data preparation

For data preparation since we are going to build 2 models (one for detection and another for classifications) it is required to perform 2 different types of data preparation. On table 4.1 we can see our final original dataset with a total of 1532 images with around 500 images per class

TABLE 4.1. Number of images per class

Class	Number of images
High Irrigation	502
Medium Irrigation	504
Low Irrigation	526

4.1.1. Data Preparation for Object Detection

For our object detection models we uploaded the dataset on roboflow following the steps on this paper[29], with the data on roboflow we are able to label the images and perform data augmentation.

It was then created 2 datasets with the same images but different labels, one of the datasets had the full labels (No stress, medium stress and high stress), the other dataset had only one label to identify the plant.

Each image on the 3 label dataset was individually and manually annotated, after that the annotations were cloned and had their data yaml files edited to possess simply 1 (plant).

The model with 3 labels although it can differentiate the water stress on each plant this prediction will not be accurate since it will simply use the RGB image and no thermal data whatsoever and this prediction will vary according to the thermal scale of the image. That said it can still detect the object and draw the bounding box around it.

The bounding box will then be used with the classification model and the thermal array to perform assess accurately the water stress of the plant.

For data augmentation we used rotations, 90 degree flips and zooms between 0% to 25%, to increase the number of training images to 3216 (it could not give us more images due to limitations on this roboflow version, too many data augmented images will also

harm our performance as it will make the model to remember the data instead of learning any meaningful patterns) and they were resized to 640x640 (following paper [29], [36] and if we check the yolo documentation [39] YOLOv11 also uses that pixel size) since a square shaped size improves model quality.

The images did not take treatments like the CLAHE [27,33] or ICE [27] to improve their contrast.

4.1.2. Data Preparation for Classification

The data preparation for classification required a different treatment since in here we did not use the image, to classify water stress, instead the thermal values embedded within the image were converted to numpy arrays to be used as input for the CNN [18].

As it was previously mentioned, we tried using Flir tools mobile app to transfer the image to from the camera to the phone but this process deleted all the thermal data since only a snapshot of the image was saved.

This information was obtained firstly by using the online version of "ExifTool". This is a powerful tool that allows us to read, write or edit meta information on a wide variety of files. "ExifTool" also supports various maker notes of many digital cameras such as "FLIR" [38].

At first the data used was from the first pictures which were taken from the FLIR E60 thermal camera and were transferred to a phone then to the computer.

The problem with this method is that after the data was processed by the "FLIR TOOLS" mobile app, only a snapshot of the image was saved which had no thermal data embedded into it. With this data an attempt to convert the pixels in the image was made by converting the image to grayscale and normalize the data with the thermal scale, but this method was inefficient and would not have 100% accurate thermal data.

As such, to overcome this issue and access the true, the pictures were taken from the camera and directly saved into SSD memory card, the existence of thermal data within these images was then confirmed with "ExifTool".

To extract this data and process it, a Python library named "FlirImageExtractor" combined with "ExifTool" locally, to extract the thermal values from the images into numpy arrays (If a library like openCV is used directly on the image, it will destroy the thermal data so the original image must be kept untouched).

After extracting the thermal data to numpy arrays, the room temperature of each photo session was subtracted to each thermal value in the array for the respective image.

For example during the photo session on day 1 the ambient temperature was of 22 degrees celsius, the leaf temperature for the plant at low stress was around 21-22 that means when we subtract the ambient temperature to it, the value of pixel should be

around -1 to 0 while for a plant at higher stress which its temperature was around 23 to 24 that value would be between 1 and 2.

This will normalize the temperature of the plant in relation to the room temperature, allowing us to relate and predict the water stress in the plants during different days, as long as the weather conditions are similar to the ones we took the pictures.

The plants with no stress will always be under room temperature while the ones at medium stress should be at around room temperature and the plants with high stress are very likely to have a temperature higher than the temperature of the room.

Class	Canopy temperature
High Irrigation	Lower than room temperature
Medium Irrigation	Around the same as room temperature
Low Irrigation	Higher than room temperature

TABLE 4.2. Relation between irrigation and temperature

With the values established when feeding the images into the models, for the custom models we simply gave them the entire numpy array with an extra batch channel making it (240,320,1) while on the models that used transfer learning, these arrays needed to be resized to a square shape of 224x224 and have their channels cloned 3 times to simulate the RGB channels, since models like the mobilenet and VGG were trained on the imageNet dataset and usually take RGB images as input, making the shape of the arrays: (224,224,3).

For data augmentation for the classification models there was also a special care as color changes and blur do not make sense when working with thermal arrays as they would damage our data. So the only augmentation that was performed were horizontal and vertical flips.

Even though the camera has a resolution of 320x240, if the pictures are taken holding the camera vertically, the JPGs will be saved as 240x320 but the thermal data will still be saved with a numpy array of (240,320) regardless. This will be important because there was another dataset version, this one was made using the yolo bounding boxes.

A third dataset was built using the bounding boxes created by the labeling process. Although this will resize our images and cause the thermal values to become an average of its neighbors this will also allow us to remove the background that was left on the images while making it more resilient when it is used in conjunction with the detection model.

By fitting the bounding boxes inside the thermal array and resizing the array to the desired shape for both model types (as it was mentioned before 224,224 for transfer learning and 240,320 for the custom models). Because the images were taken at close range(around 75 cm distance from the camera to the plant) the image will not alter a lot

since most of it is part of the plant, this way we remove the remaining noise exists and when we apply the resize it will not damage the true data by a lot and will make the data more fitting for the yolo bounding box predictions.

Because some of the images had a rotated resolution but the numpy array was kept the same, when fitting those images inside the bounding boxes it was required to verify their shape and rotate the bounding box coordinates if needed to fit inside the numpy array.

4.2. Modeling

4.2.1. Object detection

For our detection models, we decided to use the YOLO11, as it is the most recent version and the best performing according to a test on the COCO dataset and displayed by ultralytics this can be observed on figure 4.1. For the version of YOLO11 we choose YOLO11n as it is the smallest and lightest model, with 2.6(M) parameters yet it performs very well with a drop of 12 in the mAP@50-95 metric when compared with its heavier counterpart YOLO11m which has almost 10 times more parameters (20.1) this will allow our model to run on low hardware machines and simplify our training [39].

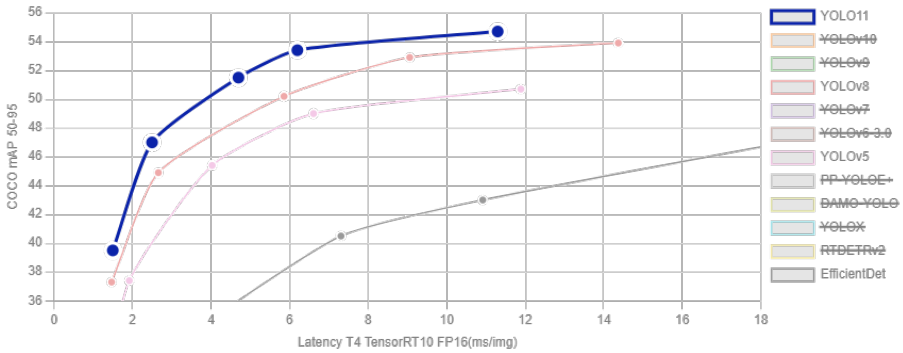


FIGURE 4.1. Comparison between YOLOv11, YOLOv8, YOLOv5 and EfficientDet

For the object detection model it was trained 2 models, they behave exactly the same when it comes to detecting plants, the only difference being that one of the models has 3 labels (No stress, Medium stress and High stress) and the other only has 1 label (Plant).

To train our models, the dataset was split into 70%, 20%, 10% (training, validation and testing) giving us 3216 train images (1072 pre augmentation), 307 validation images and 153 test images.

The batch size was set to 64, with the number of epochs set to 100 with a patience of 10, to stop the training earlier and prevent overfitting. Both models used AdamW optimizer to define the learning rate and momentum. (lr=0.002, momentum=0.9 for the single class model; lr=0.001429, momentum=0.9 for the multi class model)

4.2.2. Classification

For our classification models we performed a custom model, trained from scratch and used transfer learning training only the head for classification.

When using transfer learning with our classification models we decided to take a light-weight models due to computational limitations, as such we chose light models like the mobileNetV2 following paper[23] but we also tested the VGG16 which is supposed to perform well although heavier as it possesses a much higher amount of parameters than the mobileNetV2. We also performed a custom model on our 2 different datasets mentioned in the data preparation chapter, the normal dataset and the dataset with reduced background by using the bounding boxes made for YOLO as it was mentioned before.

The architecture used for the custom models consists of three consecutive 3x3 convolutional layers with RELU activation functions. The number of kernels increases progressively from 16 to 32 and finally to 48 and each convolutional layer is followed by a max-pooling operation to reduce spatial dimensions. After these layers the feature maps are flattened and passed to a fully connected layer with 48 neurons. To mitigate overfitting, a dropout of 0.5 was applied. Finally the network ends with a softmax activation function since we have 3 different classes, this architecture is displayed on figure 4.2.

The data was split into 80% for the training dataset and 20% validation dataset giving us 1225 training images and 307 validation images.

Each model had a batch size of 32 and the number of training epochs was set to 50 with an early stop with patience value of 5 and a learning rate of 0.001. When testing with a smaller learning rate (0.0001 or 0.00001), at the cost of a slower convergence the model showed a more stable training in the display graph, but the training results were the same so we choose to stay with the standard learning rate of 0.001.

The patience feature will monitor the validation loss value and if this value does not go lower for more than 5 epochs the training will stop and save the weights where the validation loss stopped improving, which is considered the best weights.

As we previously mentioned we chose the mobileNetV2 because of its lighter architecture when compared to other transfer learning models. We also tested a VGG16 which is a heavier model than the mobileNetV2 to compare results.

These models were trained with the same specifications as the custom models. 50 training epochs, a batch size of 32, a early stopper with a patience of 5 and a dropout of 50% as well.

What changes from this model to the custom ones, is that the transfer learning models had 128 neurons on the fully connected layer since these models are already pre-trained on the image net dataset we can increase the number of neurons without it overfitting in comparison with the custom models.

The transfer learning models also had their pre-trained body weights frozen so we could train only the classification head of the model, the architecture is displayed on figure 4.3. That said there was no fine-tuning, but simply training the head of the model presented very good results in the evaluation metrics, superior to the results on the custom models.

As for data augmentation only horizontal and vertical flips were applied, expanding the number of training images from 1225 to 3675. This is supposed to help the model have variations of the data in the dataset with the objective of increasing its ability generalize the data, allowing it to detect different patterns in plants.

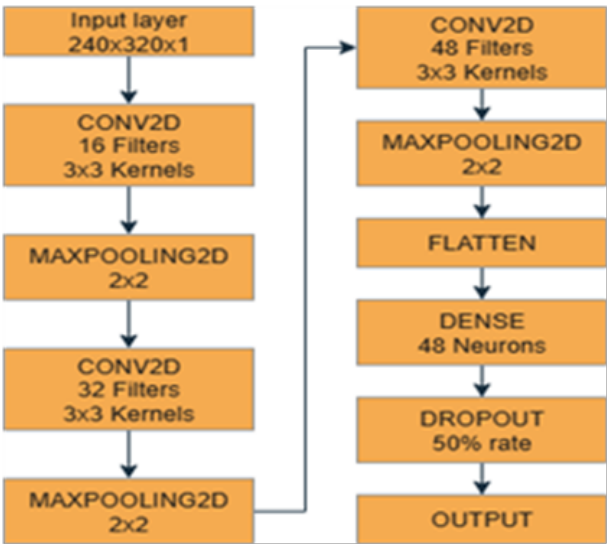


FIGURE 4.2. Custom CNN architecture

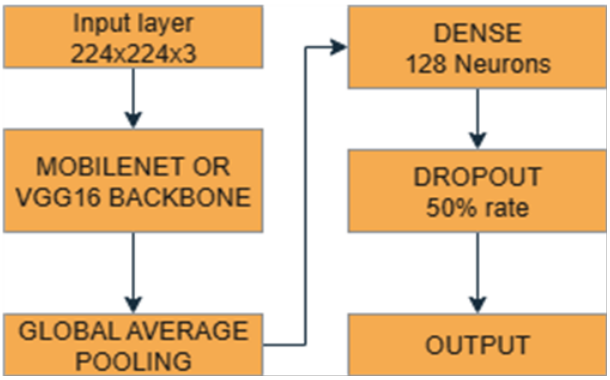


FIGURE 4.3. Transfer learning architecture

4.3. Performance evaluation metrics

4.3.1. Classification

For classification we can use metrics like precision, recall, F1-score and accuracy[18]. Since we have 3 different classes we are required to calculate the precision recall and F1-score individually for each class and then perform an average.

- Recall: $\frac{TruePositives}{(TruePositives + FalseNegatives)}$; it evaluates the true positive rate.
- Precision: $\frac{TruePositives}{(TruePositives + FalsePositives)}$; evaluates how many true positives are well predicted.
- F1-Score: $\frac{2 * (Precision * Recall)}{(Precision + Recall)}$; is a balance of the recall and precision.
- Accuracy: $\frac{(TruePositives + TrueNegatives)}{TotalItems}$; evaluates how well the model makes predictions.

These metrics are important since they will allow us to check for correlations between the predictions and our data.

4.3.2. Object detection

- P (Precision): The accuracy of the detected objects, indicating how many detections were correct.
- R (Recall): The ability of the model to identify all instances of objects in the images.
- mAP50: Mean average precision calculated at an intersection over union (IoU) threshold of 0.50.
- mAP50-95: The average of the mean average precision calculated at varying IoU thresholds, ranging from 0.50 to 0.95.

Usually the mAP@50-95 is the most used metric since it allows us to see how well does our model really detect objects across various intersection over union(IoU) thresholds (ranging from 0.50 to 0.95) and while the mAP@50 is not a bad metric, it is less rigid since it will take into account any detection with an IoU threshold of over 0.5 which are easier detections [39]. Both mAP@50 and mAP@50-95 are variants of average precision at different IoU (Intersection over Union) thresholds [25]. Intersection over union measures the overlap between a predicted bounding box and a ground truth bounding box. For our problem we will mostly take into account the mAP@50, mAP@50-95 and recall since the purpose of the object detection model is not to classify the objects, it is only to detect the objects.

4.4. Model Evaluation

4.4.1. Object detection models

The model with a single label had a recall of 1, a precision of 0.999, a m@AP50 of 0.995 (which is the less strict metric) and a m@AP50-95 of 0.875. The model with 3 labels had a precision of 0.995, a recall of 0.997, a m@AP50 of 0.995 and a m@AP50-95 of 0.915. These metrics can be observed in figure 4.4 and figure 4.5.

When comparing both models interestingly the model with 3 labels has a better performance since the IoU boxes it draws are more accurate (m@AP50-95 of 0.875 compared to 0.915), the different stress levels labelled possibly allowed it to learn more meaningful features. Although when testing both models in real data the models both performed quite well and identically. So both models are valid for detection purposes but if we have to choose using metrics the model with the 3 different labels would be the one.

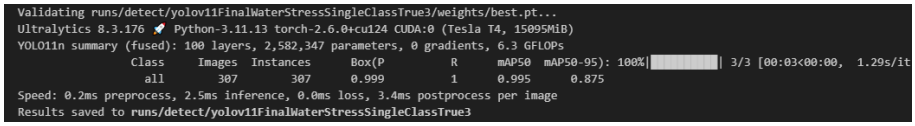


FIGURE 4.4. Yolo11n detection model metrics (single class)

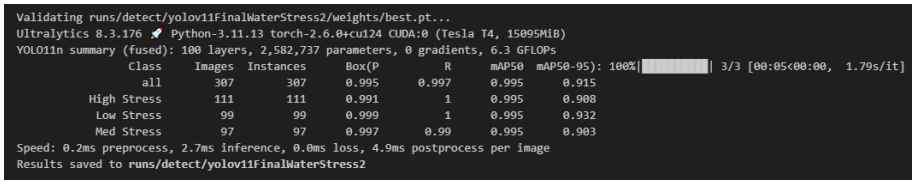


FIGURE 4.5. Yolo11n detection model metrics (multi class)

We also performed various detection tests on images that were not used for training or testing and the quality of the detections from each model is very similar as we can see on figure 4.6 and figure 4.7.

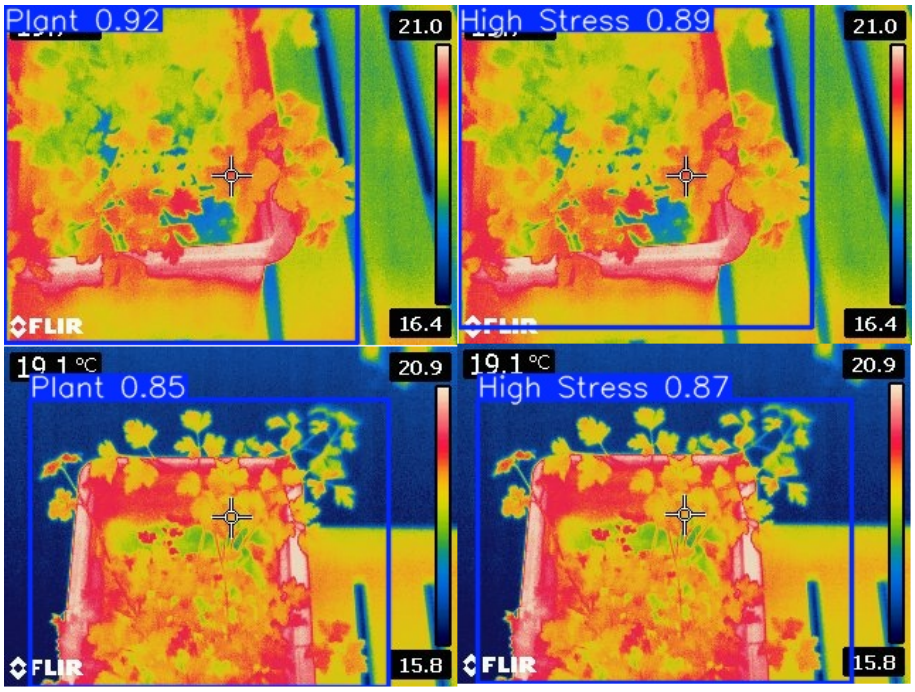


FIGURE 4.6. Prediction comparison 1

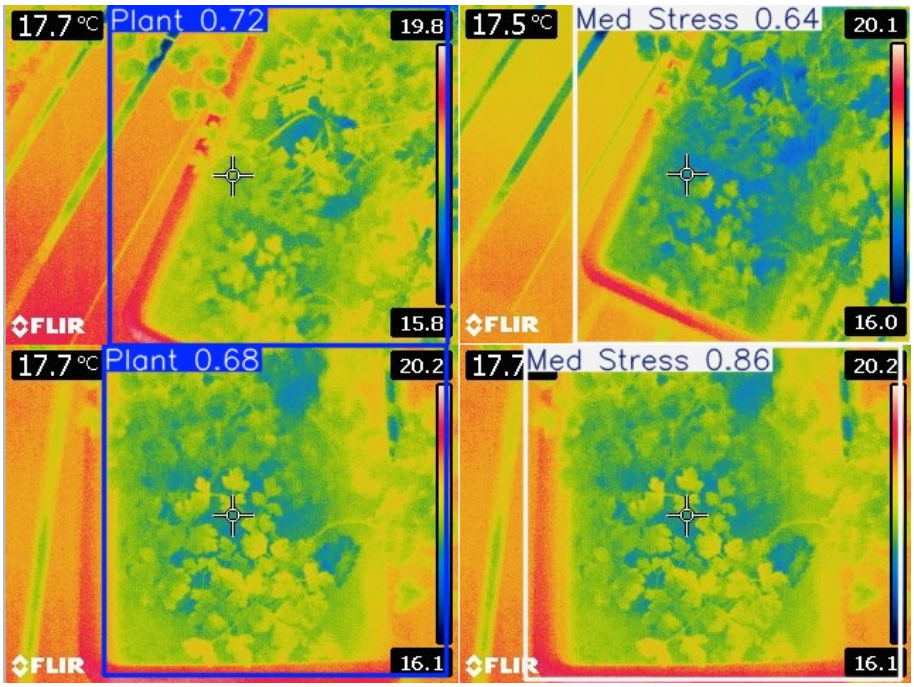


FIGURE 4.7. Prediction comparison 2

4.5. Classification models

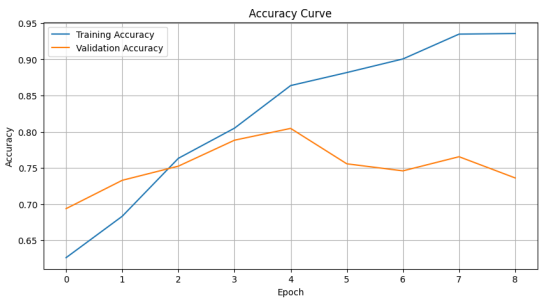
4.5.1. Custom models

The first two custom models built, were trained the original dataset one possessed data augmentation which is. Overall both models present good results, the early stopper performed his task and stopped the model before it starts to overtrain (by epoch 3 for the custom model without data augmentation on figure 4.8.(a) and by epoch 9 on the model with data augmentation on figure 4.9.(a)) where both its accuracy values and loss values start to diverge as we can observe both on figure 4.8.(a and b) and figure 4.9.(a and b), with this difference being a lot more accentuated on the model without data augmentation.

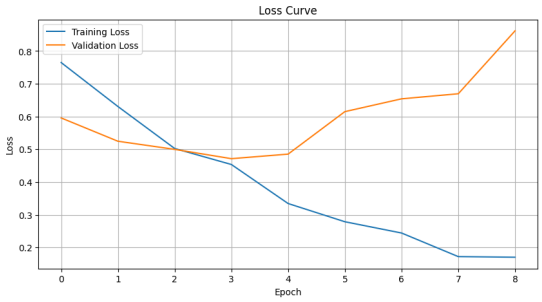
When reviewing both confusion matrices, we can observe on figure 4.8.(c) that the model without augmentation rarely attributes the no stress label to a high stress plant and never attributes a high stress label to a no stress plant, this is also observed on the confusion matrix on figure 4.9.c) where these labels are never attributed to one another.

This is likely due to the fact that both stress levels are the extreme opposites therefore and the model seems to have learned the patterns behind these extremes. On the other hand it seems to fail quite a bit when predicting the ”medium stress” class as we can see on table 4.3 where the ”medium stress” class has a precision of 0.66 and on table 4.4 where it has a precision of 0.67, likely due to the fact that this is the middle class, in a way a bridge from high water stress to no water stress. Overall, even with a bit of trouble identifying medium stress levels both models still perform well, with the model with data augmentation being better by a small margin.

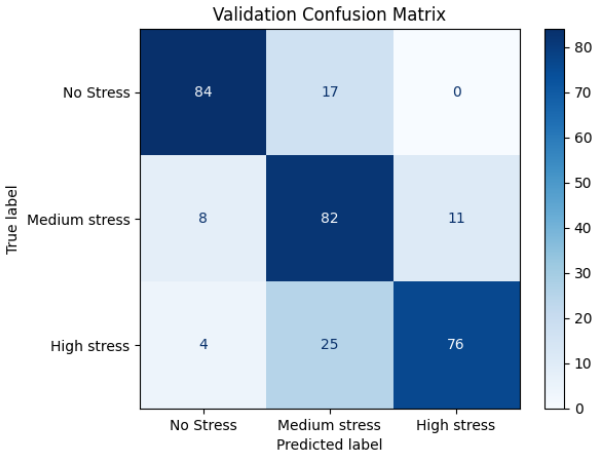
The custom model without augmentation has an accuracy of 78.53%, an average precision of 80% and average recall of 79% and the custom model with data augmentation has an accuracy of 80.74%, an average precision of 81% and average recall of 80% (these values from all models will be displayed on table 4.11 for a better all around view).



(A) Accuracy Graph



(B) Loss Function Graph)

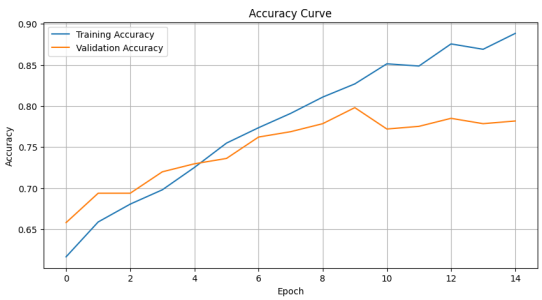


(C) Confusion matrix

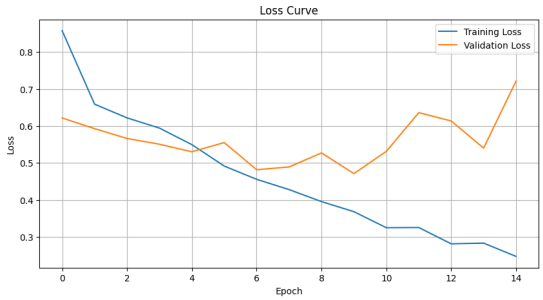
FIGURE 4.8. Custom Model without Data Augmentation

	Precision	Recall	F1-Score
No Stress	0.88	0.83	0.85
Medium Stress	0.66	0.81	0.85
High Stress	0.87	0.72	0.79

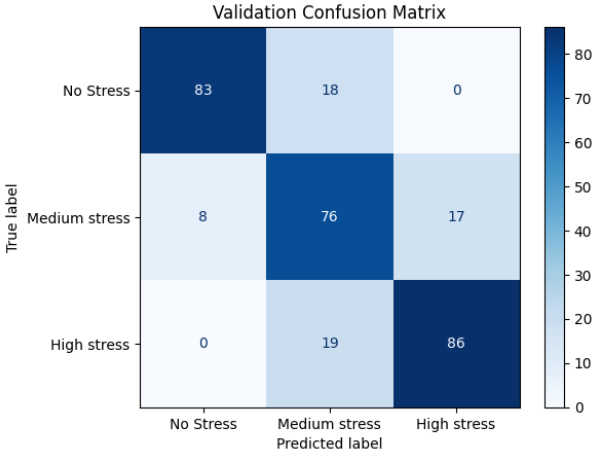
TABLE 4.3. Evaluation metrics of Custom Model without data augmentation



(A) Accuracy Graph



(B) Loss Function Graph



(C) Confusion matrix

FIGURE 4.9. Custom Model with data augmentation

	Precision	Recall	F1-Score
No Stress	0.91	0.82	0.86
Medium Stress	0.67	0.75	0.71
High Stress	0.83	0.82	0.83

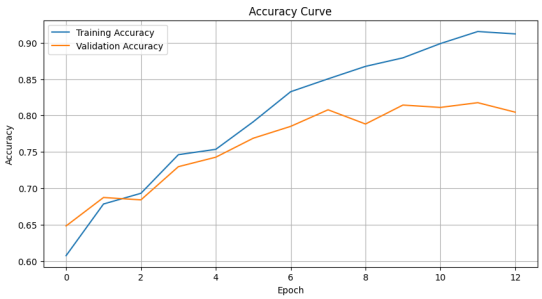
TABLE 4.4. Evaluation metrics of custom model with data augmentation

The final custom model was trained on the dataset with reduced background, where we used the bounding boxes and fit them on the numpy thermal array as it was previously mentioned on the data preparation chapter. This model had the same data augmentation as the last applied to it as it improved the performance of the previous custom model.

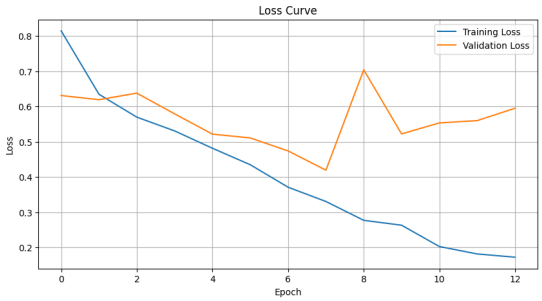
This model also presented good metrics, with results very similar to the last one with the augmented data. The training was overall stable as it is displayed on figure 4.10.(a and b) with a small spike on the loss function, just before the training ends by epoch 7. The confusion matrix on figure 4.10.(c), even with the treatment we applied to the dataset still followed the same logic as the previous ones.

With 82.86% accuracy and 80.78% validation accuracy, here we started to see s slight overfitting, an average precision of 81% and an average recall of 81%. Since the metrics

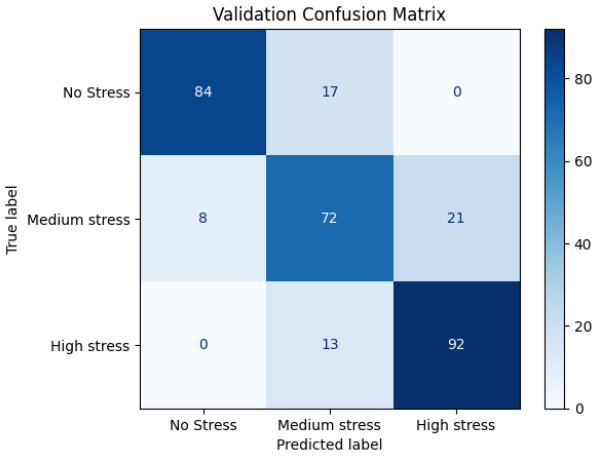
are good and similar to the metrics of the model with data augmentation, the model trained on data whose resize follows what we expect the bounding boxes will give the classification models as input, this will advantage over the previous models.



(A) Accuracy Graph



(B) Loss Function)



(C) Confusion matrix

FIGURE 4.10. Custom Model with data augmentation trained on the reduced background dataset

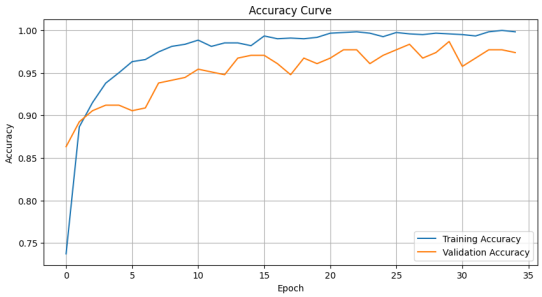
	Precision	Recall	F1-Score
No Stress	0.91	0.83	0.87
Medium Stress	0.71	0.71	0.71
High Stress	0.81	0.88	0.84

TABLE 4.5. Evaluation metrics of the custom model with data augmentation trained on the reduced background dataset

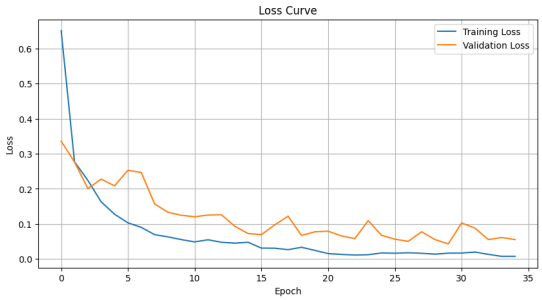
4.5.2. Transfer learning

When evaluating the transfer learning we can see that it was able to use its previous knowledge and apply it to our dataset. The training graphs on image 4.11.(a and b) show a slower yet more stable training, due to the fact that these architectures are much more complex than that of our custom model. The training stops by epoch 30, achieving an accuracy of 99.67% with a validation accuracy of 98.70%, 99% average precision and average recall which are very good metrics and showing no overfitting.

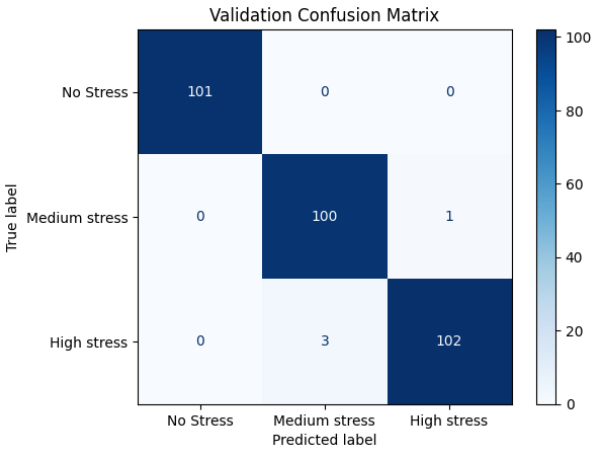
When analyzing its confusion matrix on figure 4.11.(c) and table 4.6 although at a very small scale since it predicted almost everything correctly, we can still observe that the "medium stress" class is the hardest to predict by the few miss classifications it performed, reinforcing the what was mentioned when evaluating the previous custom models.



(A) Accuracy Graph



(B) Loss Function Graph)



(C) Confusion matrix

FIGURE 4.11. MobileNetV2

	Precision	Recall	F1-Score
No Stress	1.00	1.00	1.00
Medium Stress	0.97	0.99	0.98
High Stress	0.99	0.97	0.98

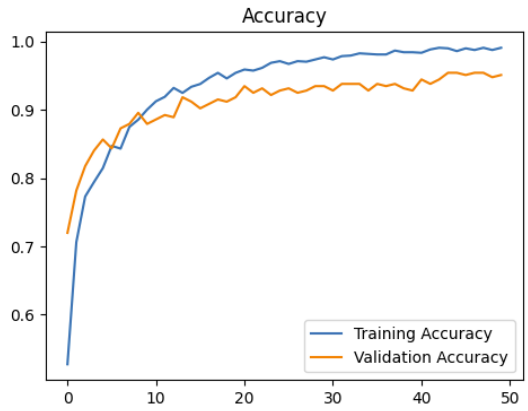
TABLE 4.6. Evaluation Metrics MobileNetV2

The VGG16 presented somewhat different results, as this model trained for all the epochs and showed small signs of overfitting, with a validation accuracy of 95.11% and a training accuracy of 99.06%.

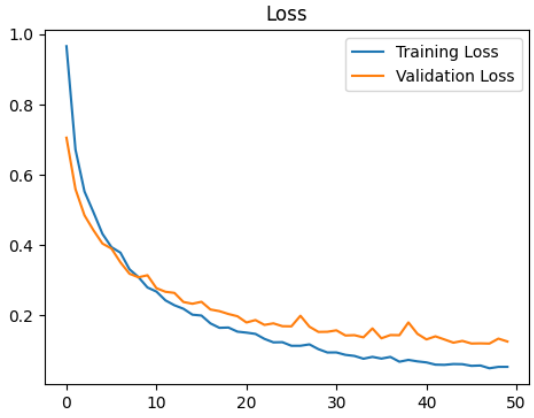
This model also took a lot longer to train and converge, not only due to the higher number of epochs (it reached training epoch number 50) but also because of the fact that it contained a lot more parameters than the mobileNetV2, with 14,780,739 total parameters on the VGG16 compared to the 2,422,339 parameters on the mobileNetV2.

We can still observe a smooth training on figure 4.12.(a and b), with some divergence which justifies the difference in validation accuracy and training accuracy and on the confusion matrix the same difficulties as the previous, with an outlier that identified a ”no stress” class as a ”high stress” class, this is demonstrated by the results on table 4.7.

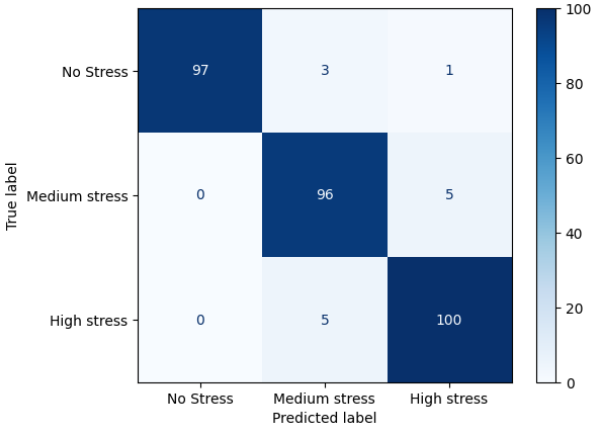
With these results in mind we decided to not proceed further with the VGG16, due to the fact that the mobileNetV2 is much lighter and presents better results.



(A) Accuracy Graph



(B) Loss Function Graph)



(C) Confusion matrix

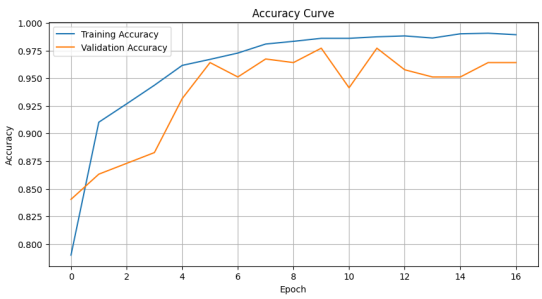
FIGURE 4.12. VGG16 model

	Precision	Recall	F1-Score
No Stress	1.00	0.96	0.98
Medium Stress	0.92	0.95	0.94
High Stress	0.94	0.95	0.95

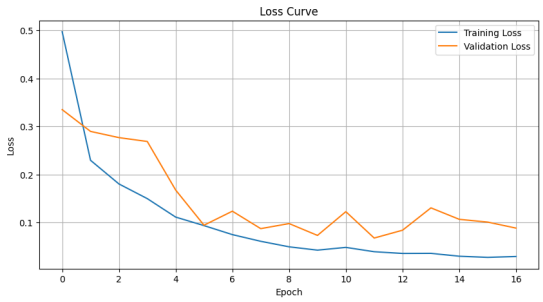
TABLE 4.7. Evaluation metrics of the VGG16 model

When applying data augmentation on the MobileNetV2 model, the metrics did not change much, since the model had near perfect metrics this is to be expected.

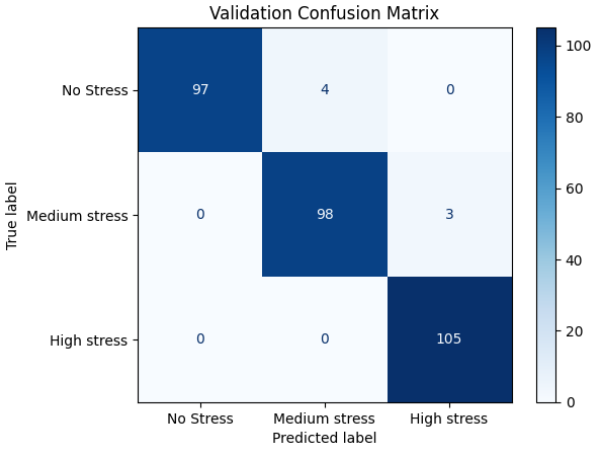
When analyzing the graph on figure 4.13.(a and b), we observe that it finished its training earlier (by epoch 11), but each epoch took more time to train since the training dataset was larger due to augmentation. On the confusion matrix on figure 4.13.(c) and the table 4.8 presented the same difficulties as previous models to classify the "medium stress" class.



(A) Accuracy Graph



(B) Loss Function)



(C) Confusion matrix

FIGURE 4.13. MobileNetV2 with data augmentation

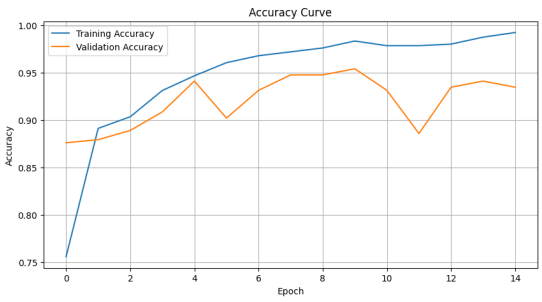
	Precision	Recall	F1-Score
No Stress	1.00	0.96	0.98
Medium Stress	0.96	0.97	0.97
High Stress	0.97	1.00	0.99

TABLE 4.8. MobileNetV2 with data augmentation

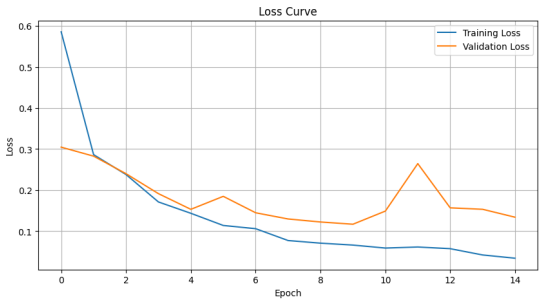
Finally we reach the final two transfer learning models which are the mobileNetV2 trained on the reduced background dataset, with and without data augmentation and both in the previous models and in the next models, data augmentation did not have any impact in these transfer learning models.

The only difference that we can observe now is that the mobileNetV2 trained on the dataset with reduced background, both with and without data augmentation presented some overfitting as it can be displayed by both on figure 4.14.(a and b) and figure 4.15.(a

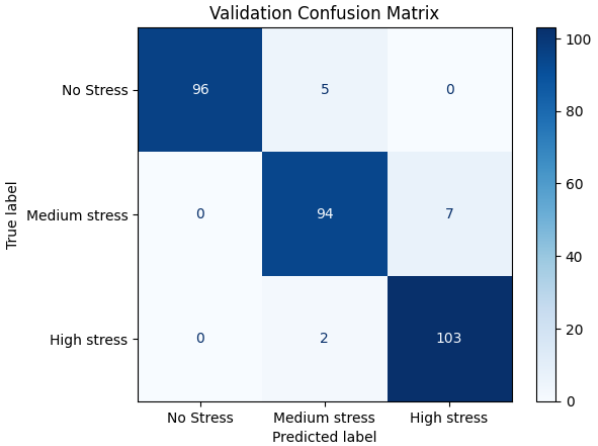
and b) as the values on these graphs start to diverge, this is more accentuated on the model without data augmentation as it finishes training in less epochs (saves weights at epoch 9 and finishes by epoch 14) and on both tables(table 4.9 and table 4.10) we can see once again difficulty in classifying the ”medium class”.



(A) Accuracy Graph



(B) Loss Function)

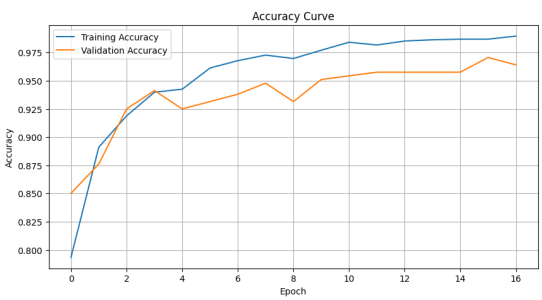


(C) Confusion matrix

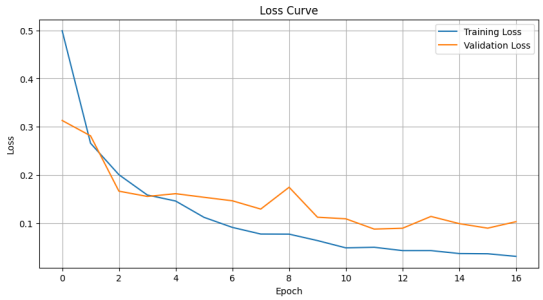
FIGURE 4.14. MobileNetV2 trained on the reduced background dataset

	Precision	Recall	F1-Score
No Stress	1.00	0.95	0.97
Medium Stress	0.93	0.93	0.93
High Stress	0.94	0.98	0.96

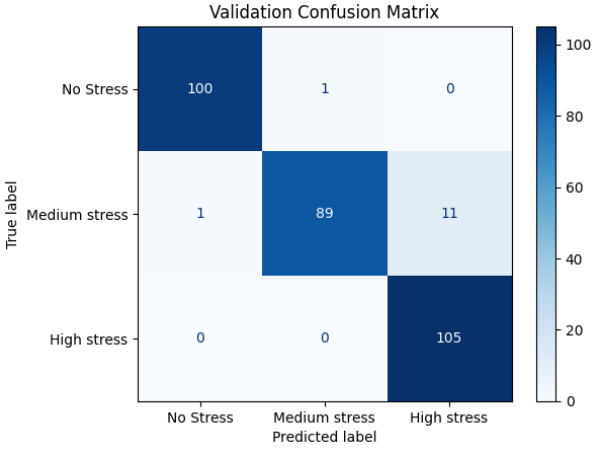
TABLE 4.9. MobileNetV2 trained on the reduced background dataset



(A) Accuracy Graph Custom Model 1



(B) Loss Function Graph Custom Model 1)



(C) Confusion matrix Custom Model 1

FIGURE 4.15. MobileNetV2 with data augmentation on reduced background dataset

	Precision	Recall	F1-Score
No Stress	0.99	0.99	0.99
Medium Stress	0.99	0.88	0.93
High Stress	0.91	1.00	0.95

TABLE 4.10. Evaluation metrics of the MobileNetV2 with data augmentation trained on the reduced background dataset

4.5.3. Model selection and discussion

	Accuracy	Validation Accuracy	Average Precision	Average Recall
Custom model	78.53%	78.83%	80%	79%
Custom model with data augmentation	80.74%	79.80%	81%	80%
Custom model with data augmentation trained on the dataset with reduced background	82.86%	80.78%	81%	81%
VGG16	99.06%	95.11%	96%	95%
MobileNetV2	99.67%	98.70%	99%	99%
MobileNetV2 with data augmentation	98.26%	97.72%	98%	98%
MobileNetV2 trained on the dataset with reduced background	98.75%	95.44%	95%	95%
MobileNetV2 with data augmentation trained on the dataset with reduced background	98.03%	95.77%	96%	96%

TABLE 4.11. Metrics per model

Overall among all models there was a trend where the "medium stress" class was the most challenging to classify and by observing table 4.11, we can see that data augmentation improved the quality of the custom models by a small amount and did not change the mobileNetV2.

This is possibly due to the fact that vertical and horizontal flips do not add a lot of variation to the dataset, it simply creates mirrored versions of the thermal array. But due to the limited augmentation we can perform on thermal array, it is still worth it for an improvement on the custom models.

When comparing all the metrics on table 4.11, it is observable that the transfer learning models outperform the custom models by quite a lot, with an increase of 14% to 20% across all metrics.

Although the VGG16 is an heavier model the MobileNetV2 still outperformed it.

As such for the purpose of classification the MobileNetV2 models overcome all others, but when selecting between them although the one trained on the original dataset has a better performance than the one trained on the dataset with reduced background, I believe we should choose the model trained on the reduced background dataset, since the data it trained on was resized and when applied with the detection model, the bounding boxes it detects will be given to the classification model.

CHAPTER 5

Conclusions and Future work

5.1. Conclusions

In this project, we demonstrate that combining the RGB information with the true thermal data extracted from the images substantially improves the reliability and accuracy of water stress detection and classification. The YOLO11n although light, it successfully localized the plants using only the thermal images RGB information achieving a m@AP50-95 of up to 91.5% and performed very well on the tests, confirming that visual cues alone are sufficient for the identification stage.

For the classification task, however, the integration of precise temperature values was necessary due to the fact that different thermal scales will have an impact in the classification of water stress while the true unaltered thermal values will not have that bias. We achieved and accuracy of up to 99% using transfer learning and 82.26% with custom models.

To obtain this thermal information, we applied a pre-processing pipeline using "ExifTool" and "FlirImageExtractor" to extract the raw temperature data embedded within the thermal images. These values were then converted into NumPy arrays and normalized using the ambient room temperature. This normalization step significantly enhanced the model's robustness, ensuring consistent performance across days with varying temperatures.

One of the disadvantages of the current approach was the fact that values like humidity, wind speed and sunlight were not taken into account. Because of this the model's classification abilities is limited to dry weather under shade.

Overall, the results indicate that leveraging both visual and thermal modalities as the potential to provide more comprehensive representation of plant physiological status. This multimodal approach enables more accurate detection and classification of water stress levels, highlighting its potential for practical applications in precision agriculture when implemented with aerial drones, as it was mentioned in the introduction, it allows for an agile, swift and scalable method of water stress assessment.

All research questions were answered, for the first question, we took advantage of the embedded thermal data within the image as we were able to directly input the thermal values into our CNN, allowing us to classify water stress more accurately.

For the second question, the YOLOv11n worked very well when detecting the plants, in this stage we were able to perform more augmentation to make the model more robust since it will take advantage of the RGB image and not the thermal array(the original image must still be kept intact so in this stage copies were created, allowing us to work freely on the image without damaging the metadata as it was used with the CNN for the classification of the water stress).

As for the third question, the experiments conducted using the moisture sensor to keep the field on a certain water percentage threshold seems to be the best method to achieve the desired water stress, as it is more accurate than weighting the soils specially since we have to take into account the plants biomass growing overtime in conjunction with the loss of water due to transpiration and the evaporation of moisture of the soil making this method harder to keep track of the soil moisture level and to achieve accurate plant stress levels.

5.2. Future work

For future work, it is important to consider the influence of external environmental conditions—such as solar radiation, wind speed, humidity, and growth stage which significantly affect plant temperature and stress responses. Collecting these variables over a longer period and incorporating them as inputs with the thermal values on our deep learning method or a machine learning approach, would allow the development of more robust predictive models.

To achieve this, the experiment should be replicated using a higher-precision soil moisture sensors and an automated irrigation system capable of maintaining a consistent soil moisture threshold and it should be conducted outdoors to obtain data under real circumstances rather than a controlled environment.

Additionally, implementing established physiological measurement techniques, such as the Crop Water Stress Index (CWSI) or stomatal conductance measurements using diffusion porometers, would provide more accurate ground-truth water stress values and validate these results, creating a higher-quality dataset.

Finally with the new, larger and more robust dataset will allow us train our detection model using more larger YOLO11 versions like if needed like the YOLO11m and make our classification models a lot more robust to weather conditions.

After fully training the models we should be able to implement the jetson orin which is a very compact yet powerful computer on aerial drones, allowing the models to assess water stress on large fields in an agile and swift manner, promoting the replacement of soil moisture sensors and other labor intensive methods.

References

[1] Duarte-Galvan, C., Romero-Troncoso, R., Torres-Pacheco, I., Guevara-Gonzalez, R., Fernandez-Jaramillo, A., Contreras-Medina, L., Carrillo-Serrano, R., & Millan-Almaraz, J. (2014). FPGA-Based smart sensor for drought stress detection in tomato plants using novel physiological variables and discrete wavelet transform. Sensors, 14(10), 18650–18669. https://doi.org/10.3390/s141018650

[2] De Almeida, A. M., Coelho, R. D., Da Silva Barros, T. H., De Oliveira Costa, J., Quiloango-Chimarro, C. A., Moreno-Pizani, M. A., & Farias-Ramírez, A. J. (2022). Water productivity and canopy thermal response of pearl millet subjected to different irrigation levels. Agricultural Water Management, 272, 107829. https://doi.org/10.1016/j.agwat.2022.107829

[3] Kurunc, A., Tezcan, N. Y., & Bimurzayev, N. (2023). Determination of Water Stress in Wheat Crops by Thermal Images Under Mediterranean Conditions. Communications in Soil Science and Plant Analysis, 54(12), 1713–1726. https://doi.org/10.1080/00103624.2023.2211089

[4] Han, M., H. Zhang, K. C. DejongeComas, and L. H. Trout. 2016. Estimating maize water stress by standard deviation of canopy temperature in thermal imagery. Agricultural Water Management 177:400–09. doi:10.1016/J.Agwat.2016.08.031.

[5] Aragão, M. F., Neto, L. G. P., De a Viana, T. V., Manzano-Juarez, J., Lacerda, C. F., Costa, J. D. N., Lima, J. S., & Azevedo, B. M. (2023). Evaluation of crop water status of melon plants in tropical semi-arid climate using thermal imaging. Revista Brasileira De Engenharia Agr´ıcola E Ambiental, 27(6), 447–456. https://doi.org/10.1590/1807-1929/agriambi.v27n6p447-456

[6] de O. Maia Júnior, S., de Andrade, J.R., Silva, P.C. et al. Soybean Seed Priming with Brassinosteroids Mitigates the Effects of Drought Stress. J Plant Growth Regul (2024). https://doi.org/10.1007/s00344-024-11588-1

[7] Belfiore, N., Vinti, R., Lovat, L., Chitarra, W., Tomasi, D., De Bei, R., Meggio, F., & Gaiotti, F. (2019). Infrared thermography to estimate vine water status: optimizing canopy measurements and thermal indices for the varieties merlot and moscato in northern Italy. Agronomy, 9(12), 821. https://doi.org/10.3390/agronomy9120821

[8] Ma, S., Liu, S., Gao, Z., Wang, X., Ma, S., & Wang, S. (2024). Water deficit Diagnosis of winter wheat based on thermal infrared imaging. Plants, 13(3), 361. https://doi.org/10.3390/plants13030361

[9] Gutíerrez-Gordillo, S., Garc´ıa-Tejero, I. F., Zuazo, V. H. D., Escalera, A. G., Gil, F. F., Amores-Ag¨uera, J. J., Rodr´ıguez, B. C., & Hern´andez-Santana, V. (2020). Assessing the Water-Stress Baselines by Thermal Imaging for Irrigation Management in Almond Plantations under Water Scarcity Conditions. Water, 12(5), 1298. https://doi.org/10.3390/w12051298

[10] Lee, A., Kim, S., Hong, S., Han, Y., Choi, Y., Kim, M., Yun, S. K., & Kim, G. (2019). Phenotypic analysis of fruit crops water stress using infrared thermal imaging. Journal of Biosystems Engineering, 44(2), 87–94. https://doi.org/10.1007/s42853-019-00020-2

[11] Takács, S., P´ek, Z., Cs´anyi, D., Daood, H. G., Szuvandzsiev, P., Palot´as, G., & Helyes, L. (2020). Influence of water stress levels on the yield and lycopene content of tomato. Water, 12(8), 2165. https://doi.org/10.3390/w12082165

[12] Silva, A. D. N., Ramos, M. L. G., Ribeiro, W. Q., Junior, Da Silva, P. C., Soares, G. F., Casari, R. a. D. C. N., De Sousa, C. a. F., De Lima, C. A., Santana, C. C., Silva, A. M. M., & Vinson, C. C. (2023). Use of thermography to evaluate alternative crops for Off-Season in the Cerrado region. Plants, 12(11), 2081. https://doi.org/10.3390/plants12112081

[13] Celiktopuz, E. (2023). Determination of drought tolerance of different strawberry genotypes. PeerJ, 11, e14972. https://doi.org/10.7717/peerj.14972

[14] Renó, V., Cardellicchio, A., Romanjenko, B. C., & Guadagno, C. R. (2024). AI-assisted image analysis and physiological validation for progressive drought detection in a diverse panel of Gossypium hirsutum L. Frontiers in Plant Science, 14. https://doi.org/10.3389/fpls.2023.1305292

[15] Kurumayya, V. (2023). Infrared imaging indices for genotype screening in plant drought responses. Acta Physiologiae Plantarum, 45(11). https://doi.org/10.1007/s11738-023-03604-w

[16] Parihar, G., Saha, S., & Giri, L. I. (2021). Application of infrared thermography for irrigation scheduling of horticulture plants. Smart Agricultural Technology, 1, 100021. https://doi.org/10.1016/j.atech.2021.100021

[17] Orejuela-Romero, J., Chipantiza-Masabanda, J., Carrera-Oscullo, P., & Salguero-Cajo, A. (2023). Effect of deficit irrigation on Helianthus annuus L. plants in containers. Revista De La Facultad De Agronom´ıa, 40(2), e234012. https://doi.org/10.47280/revfacagron(luz).v40.n2.02

[18] Thukral, R., Aggarwal, A. K., Arora, A. S., Dora, T., & Sancheti, S. (2023). Artificial intelligence-based prediction of oral mucositis in patients with head-and-neck cancer: A prospective observational study utilizing a thermographic approach. Cancer Research Statistics and Treatment, 6(2), 181–190. $https://doi.org/10.4103/crst.crst_332_22$

[19] Liyanarachchi, R., Wijekoon, J., Premathilaka, M., & Vidhanaarachchi, S. (2023). COVID-19 symptom identification using Deep Learning and hardware emulated systems. Engineering Applications of Artificial Intelligence, 125, 106709. https://doi.org/10.1016/j.engappai.2023.106709

[20] Khosa, I., Raza, A., Anjum, M., Ahmad, W., & Shahab, S. (2023). Automatic Diabetic foot ulcer recognition using Multi-Level Thermographic Image data. Diagnostics, 13(16), 2637. https://doi.org/10.3390/diagnostics13162637

[21] Ibrahim, A.; Anayi, F.; Packianather, M. New Transfer Learning Approach Based on a CNN for Fault Diagnosis. Eng. Proc. 2022, 24, 16. https://doi.org/10.3390/IECMA2022-12905

[22] Colaco, S. J., Kim, J. H., Poulose, A., Van, Z. S., Neethirajan, S., & Han, D. S. (2022). Pig Treatment Classification on Thermal Image Data using Deep Learning. 2022 Thirteenth International Conference on Ubiquitous and Future Networks (ICUFN), 8–11. https://doi.org/10.1109/icufn55119.2022.9829713

[23] Baran, K. (2023). Smartphone thermal imaging for stressed people classification using CNN+MobileNetV2. Procedia Computer Science, 225, 2507–2515. https://doi.org/10.1016/j.procs.2023.10.242

[24] Memari, M., Shekaramiz, M., Masoum, M. a. S., & Seibi, A. C. (2024). Data fusion and ensemble learning for advanced anomaly detection using Multi-Spectral RGB and thermal imaging of small wind turbine blades. Energies, 17(3), 673. https://doi.org/10.3390/en17030673

[25] Q. Ashfaq and M. Usman Akram, "Convolutional Neural Network Based Thermal Image Classification," 2022 2nd International Conference on Digital Futures and Transformative Technologies (ICoDT2), Rawalpindi, Pakistan, 2022, pp. 1-6, doi: 10.1109/ICoDT255437.2022.9787443.

[26] Baghezza, R., Bouchard, K., Bouzouane, A., & Gouin-Vallerand, C. (2021). Profile recognition for accessibility and inclusivity in smart cities using a thermal imaging sensor in an embedded system. IEEE Internet of Things Journal, 9(10), 7491–7509. https://doi.org/10.1109/jiot.2021.3127137

[27] Stancic, I., & Saric, E. (2024). Increasing the model classification accuracy of thermal images. 2022 7th International Conference on Smart and Sustainable Technologies (SpliTech), 1–6. https://doi.org/10.23919/splitech61897.2024.10612393

[28] Jiang, C., Ren, H., Ye, X., Zhu, J., Zeng, H., Nan, Y., Sun, M., Ren, X., & Huo, H. (2022). Object detection from UAV thermal infrared images and videos using YOLO models. International Journal of Applied Earth Observation and Geoinformation, 112, 102912. https://doi.org/10.1016/j.jag.2022.102912

[29] W. Guettala, A. Sayah, L. Kahloul and A. Tibermacine, "Real Time Human Detection by Unmanned Aerial Vehicles," 2022 International Symposium on iNnovative Informatics of Biskra (ISNIB), Biskra, Algeria, 2022, pp. 1-6, doi: 10.1109/ISNIB57382.2022.10075707.

[30] Y. Huangfu, L. Campbell and S. Habibi, "Temperature Effect on Thermal Imaging and Deep Learning Detection Models," 2022 IEEE Transportation Electrification Conference & Expo (ITEC), Anaheim, CA, USA, 2022, pp. 185-189, doi: 10.1109/ITEC53557.2022.9813980

[31] Lv, C., Mittal, U., Madaan, V., & Agrawal, P. (2024). Vehicle detection and classification using an ensemble of EfficientDet and YOLOv8. PeerJ Computer Science, 10, e2233. https://doi.org/10.7717/peerj-cs.2233

[32] Thaker, K., Chennupati, S., Rawashdeh, N., & Rawashdeh, S. A. (2023). Multispectral deep neural network fusion Method for Low-Light object detection. Journal of Imaging, 10(1), 12. https://doi.org/10.3390/jimaging10010012

[33] Wen, M., Li, C., Xue, Y., Xu, M., Xi, Z., & Qiu, W. (2024). YOFIR: High precise infrared object detection algorithm based on YOLO and FasterNet. Infrared Physics & Technology, 105627. https://doi.org/10.1016/j.infrared.2024.105627

[34] E. Sousa, K. O. S. Mota, I. P. Gomes, L. Garrote, D. F. Wolf and C. Premebida, "Late-Fusion Multimodal Human Detection Based on RGB and Thermal Images for Robotic Perception," 2023 European Conference on Mobile Robots (ECMR), Coimbra, Portugal, 2023, pp. 1-6, doi: 10.1109/ECMR59166.2023.10256301

[35] Hou, Z., Sun, Y., Guo, H., Li, J., Ma, S., & Fan, J. (2022). M-YOLO: an object detector based on global context information for infrared images. Journal of Real-Time Image Processing, 19(6), 1009–1022. https://doi.org/10.1007/s11554-022-01242-y

[36] H. Aboalia, S. Hussein and A. Mahmoud, "Infrared Multi-Object Detection Using Deep Learning," 2024 14th International Conference on Electrical Engineering (ICEENG), Cairo, Egypt, 2024, pp. 175-177, doi: 10.1109/ICEENG58856.2024.10566390.

[37] D. Bouallal et al., "Segmentation of plantar foot thermal images: application to diabetic foot diagnosis," 2020 International Conference on Systems, Signals and Image Processing (IWSSIP), Niteroi, Brazil, 2020, pp. 116-121, doi: 10.1109/IWSSIP48289.2020.9145167.

[38] $https://exiftool.org/$

[39] $https://docs.ultralytics.com/pt/models/yolo11/\#performance-metrics$

[This page is intentionally left blank.]

Article

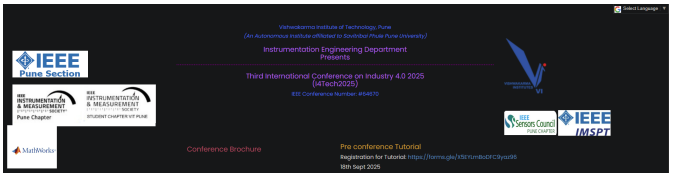


FIGURE 5.1. IEEE i4tech front page

Thermal Image Analysis for Precision Agriculture

Andre Simoes Novo
DCTI, ISTA, Iscte-Instituto Universitario de Lisboa, Instituto de Telecomunicacoes
Lisbon, Portugal
andre_novo@iscte-iul.pt

Octavian Postolache
DCTI, ISTA, Iscte-Instituto Universitario de Lisboa, Instituto de Telecomunicacoes
Lisbon, Portugal
opostolache@lx.it.pt

Joaquim Gabriel Magalhães Mendes
FEUP - Faculdade de Engenharia da Universidade do Porto, Departamento de Engenharia Mecânica
Porto, Portugal
jgabriel@fe.up.pt

***Abstract*— As global population grows, we require better management of resources, particularly an efficient water usage. This paper presents an AI solution based on deep learning for automated and non-invasive water stress detection and prediction in plants using thermal imagery. A two stage detector combining a YOLO object detector for plant location with a Convolutional Neural Network (CNN) to classify the stress level using the actual thermal values in the image and predict its stress level was implemented. These models were trained on a controlled dataset of parsley plants, with each pot on a different moisture level ("no stress", "medium stress" and "high stress"). Our system achieves a mAP50 of 99.5% for localization and the CNN had a water stress classification accuracy of up to 99.6% using transfer learning and 82.86% with the custom model. This approach proves that non-contact thermal imaging can rapidly assess plant water needs and by providing this stress-level data, it can assist farmers irrigating their crops more precisely without the need of ground sensors, saving water in the process.**

Keywords—Thermal Imaging; Plant water stress; Deep learning; Precision agriculture; Object detection; Convolutional Neural Networks; Artificial Intelligence

I. INTRODUCTION

Water is an essential resource and with the rapid rise of population growth. The total amount of water will tend to dimmish since it's a limited resource. This problem with the fact that drought is one of the most harmful abiotic factors to plants leads to the necessity of optimizing its consumption not only to save water but also to maximize the crop yields [1,2,4]. Thermography is one of these solutions. It has been used to monitor the water status of various plants [7], it's a non-invasive and non-destructive agricultural practice that is able to support the growing demand for efficiency in food production and water control. This technique allows the farmer to monitor soil salinity stress, crop diseases, yield estimation, maturity evaluation, bruise detection and water stress in plants due to the different types of radiation emoted by these factors. This makes thermal imaging a relevant addition to the field of agriculture that does not damage or provoke changes in the soil and possess swift implementation in the fields and fast accurate monitoring of the crops unlike traditional soil and plant-based methods[7], it can be expensive due to the high price of high-resolution thermal cameras but in larger fields, traditional methods are also expensive, can be difficult to set up and are more time consuming [1,2,7].

The thermographic images allow for the extraction of detailed radiometric measurements concerning temperature fluctuations, obtained through the thermal radiation emitted from the leaves, such variations occur when the plant is under different amounts of water stress due to the closure of the stomata, the higher the water stress that it affecting it, the more closed the stomata will be decreasing its transpiration and increasing its leaf temperature and this allows us to differentiated between a stressed plant and a non-stressed plant through thermal imaging[1,2].

Smart farming includes collecting accurate data on crop conditions and cultivation environments to form big data systems [6], this data can then be used with artificial intelligence to better monitor the growth and development of crops. Among this data there are thermal images. These images can be used with deep learning techniques like the YOLO, EfficientDet [22,23], Convolutional Neural Network (CNN) custom or pretrained ones like the VGG19 or MobileNet[28], for the purpose of detection and classification of objects in the case of the object detectors, or simply classification when using classifiers, all of the identifications will be based on the identification of thermal patterns. These applications are not only in agriculture but also health in the classification of patients with covid-19, to detect pedestrians in the street or to detect and differentiate pedestrians from dogs and cars [18,19,21].

By incorporating the use of a custom convolutional neural network (CNN) with an object detection model, this project aims to build a two staged detector, where in the first stage a YOLO model will to locate the plant and then will take the bounding box input and fit it inside the numpy array with the thermal values subtracted by the ambient temperature before hand and then the CNN will take this input and classify the level of water stress in the plant.

By combining thermal images with machine learning models, we can monitor and control its water conditions without the installation of a physical soil moisture sensor.

The dataset was built with the help of an E60 thermal camera to take pictures of 3 plants of parsley under different amounts of soil moisture.

II. LITERATURE REVIEW

1) Plant water stress and its biology

To build the dataset knowledge in the field of horticulture, and the impact of water stress in plants. On this paper, there was a focus on the analysis of water stress through thermal imaging.

A plant's main physiological process is photosynthesis which converts light into chemical energy and transpiration which regulates a plant's temperature through the usage of water that escapes into the atmosphere through pores in its leaf called the stomata. That being mentioned, when under drought its water reserves are lower depending on the water stress level it is at, and these levels can be divided into three stages:

Stage1(mild water stress), in this stage there is a reduction of the transpiration caused by a decline of stomatal

FIGURE 5.2. Page 1 of the article

conductance, which will force the plant to close its stomata, reducing its transpiration.

Stage2(Moderate water stress): Here the stomatal conductance is further decreased and there will be significant decrease in the photosynthetic activity of the plant.

Stage3(severe water stress): In this stage the stomatal conductance drops below its threshold value, impairing its photosynthetic capacity and causing permanent damage to its photosystems [8].

Water stress will hinder the plants' growth and can sometimes be viewed with the naked eye through the wilting of the plant, other times only other tools like moisture sensors, pressure chambers or leaf diffusion porometers allow us to monitor its water stress level [7,8].

When using thermal cameras, a simple way to identify water stress is by checking the difference between canopy temperature and air temperature, it does not take into account external factors like wind, solar radiation or humidity unlike the crop water stress index but it also does not require reference values [3,5,17].

Another thing to note is that different plants possess different temperatures when under the same level of stress, for example the tree Pometia tomentose is of 31.8ºC while in the Mezzettipsis creaghii it was of 21.6ºC or the common bean whose temperature is at 34ºC under heavy stress [1,10]. Not only from plant to plant that this temperature variation occurs but different genotypes within the same plant also possess different temperature variations for example the strawberry, the cotton plant, beans or chickpeas [11,12,13].

Another study conducted by the authors used moisture sensors to check the soil moisture capacity and performed 12 different treatments with irrigation ranging from 100% of the crop's maximum evapotranspiration to 40% on maize [2], the evapotranspiration method was also used in [3]. Another similar experiment [1] with 4 different irrigation levels (100%,75%,50%,25%) on wheat crops pre irrigation on March 08 and April 25 and post irrigation on March 10 and April 26. The temperature for mild and moderate water stress wasn't very much but there was a noticeable difference from severe to full irrigation, 3,4 ºC minimum and 6 ºC maximum on both march 08 and April 25, the emissivity on the thermal camera was between 0.97 to 0.99. In [9] the authors used a scholander pressure chamber with the stem water potential during the hottest hours of the day, instead of a moisture sensor to maintain the desired moisture level.

Another test we can perform is watering the plant and analyzing the drought effect over time [4], use fixed irrigation treatments [13,14] or irrigating only with the plants water demands and reducing these values to induce stress (100%, 75%,50%) [16,17].

One thing to keep in mind is that cold weather will impact plants temperature, due to lower temperatures there will be a lower transpiration rate this will lead to limited use of thermal imagery during these conditions [5].

2) Modeling on thermal imaging:

For the machine learning research there was a focus on both object detection like the YOLO which is a one stage detector very good for real time detection due to its speed but low accuracy for small objects[22] and classification models to recognize and classify the image with great accuracy and precision in our case, applying thermal images as input, these classification models can be trained on pre-trained models like the VGG-19 which was trained on 1.2 million general objects from the imageNet dataset[28] or simply custom built CNN[36,37].

2.1 Classification:

A convolutional neural network (CNN) is an artificial neural network used to recognize and classify images as it was mentioned before. When building a CNN there are 3 main layers to extract features. There is the convolutional layer, max pooling and finally the fully connected layer. The convolutional are used to extract the features through kernel filters applied usually on images to extract specific patterns by sliding through the input data, performing multiplications and additions generating the feature map in the process. The pooling layer is used to decrease the size of each feature map built by the convolutional layers, decreasing the number of learned parameters and the performed computations in the network. The output of these convolutional layers and poolings will then be flattened to a one-dimensional array and connected to the fully connected dense layer linking every input to to every output. The number of nodes on the last fully connected layer must be the same as the number of classes we are trying to classify. Besides the 3 main layers there is also the dropout layer which randomly ignores a specific percentage of neurons during training to help prevent overfitting and the activation function. This paper for example uses 1 layer with 32 3x3 kernels and 2 layers with 64 3x3 kernels with 2x2 max pooling layers in between each convolutional layer.[37]

If the images we have are not enough there is a process called data augmentation that generates more images to increase the size of the dataset through the creation of variations of existing images using rotations, horizontal and vertical flips, zoom in and out. For example, through 500 images we can get 1500 by using 3 different rotations (90º, 180º, 270º) [24]. There are some techniques like CLAHE and ICE enhance the contrast on thermal images, making the objects a lot more visible [20,25].

If we don't have a lot of data a pre-trained model can be implemented to leverage the resources needed to train a new model, such models can be a VGG-16, Efficient-Net, ResNet and Inception-V3. Although powerful the input data needs to be resized to 224x224 and requires 3 channels (RGB) [28]. The images can also be resized to lower resolutions like it was done in this paper [30] where they were resized to 112x112 and trained for 100 epochs with the default learning rate of 0.001 and a split train test validation of 60/20/20(although it can also be split into 80/20 train test [35]) using 4 different models, LeNet, AlexNet, VGGNet and their custom model. There is also the MobileNetV2 which is very lightweight and performs quite well [33,26].

When using the image values gaussian noise can make the model resilient to blur [34].

2.2 Object detection:

Object detectors like YOLO can be used without any training (since these were already pretrained) and they are already able to detect some objects[31], but for better results it is recommended to train on a custom dataset to adapt it to our data, the model also trains better on square shaped images so resizing them to 640x640 is recommended [29]. These algorithms locate the object by determining their size and position in the image to then classify them. To train an object

FIGURE 5.3. Page 2 of the article

detection model first we are required to have bounding boxes on the objects we want to detect and then label the bounding box for it to classify them. The evaluate the precision, recall and mAP(mean average precision), this last metric is calculated across different IoU(Intersection over Union) which indicates how close is the predicted bounding box to the ground truth, if this score exceeds a certain threshold it is considered a successful detection in particular the mAP@0.5-0.95 was the average of different IoU from 0.5 to 0.95 [21,22]. Another way to achieve better results is to use two models and accept the prediction of the model with the highest confidence [23], fusion methods that use both thermal and RGB images [19,32] or using image segmentation to detect and classify which pixel belongs to what in the image [27]. Data augmentation and methods like CLAHE or ICE can also be applied on object detection.

III. Experiments and Data Acquisition

To build our custom dataset we simulated different irrigation conditions on parsley plants, to achieve 3 classes of different water stress levels (High water stress at 100% soil moisture, medium at 50% soil moisture and low at 25% soil moisture) and the images were taken with a FLIR E60 thermal camera.

3.1) Experiments:

The first experiment was conducted under very cold conditions and was the least accurate one in terms of water measurements. There were 6 parsley plants, 2 of which were watered once per week, 2 were twice per week and the last 2 3 times per week, all of them were irrigated with 50ml of water per session. During this process their thermal values were checked but with no substantial difference between each other even though the plants that were being watered once a week were very wilted.

The second experiment was conducted during warmer weather conditions; the last 6 plants were nourished back to health and grouped in pairs on 3 different bigger pots each with 2kg of soil. Then to help them grow and acquire more leaves during the first month a fertilizer was added to their water (about 2ml per Liter of water). After that, a soil sample was measured without water, then irrigated until saturation, some incisions were made on the container to drain the saturated amount of water, and it was weighted again. The full amount was considered 100% and the dry amount was 0% after some conversions we were left with 3768.6g for the pot at 100%, 3054.3g for 50% and 2697.1g for 25%. This method possessed a flaw, the conversion could be made if the soil was heated to the point where the water would evaporate to be at exactly 0%, so this soil moisture at 50% and 25% was incorrect and the pot at 100% was experiencing root rot due to excessive water.

The final experiment, which used a soil moisture sensor to achieve much better accuracy, confirmed the last theory within the second experiment. After adjusting the soil moisture, the pot at 25% was at around 1800g, the one at 50% was at 2300g and although the pot at 100% was previously at the correct weight, its water percentage was lowered by 15% to avoid root rot.

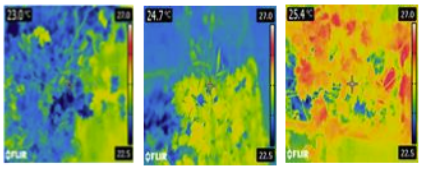
To improve accuracy, the final experiment employed a soil moisture sensor to directly measure water content within the soil. Based on its feedback, moisture levels were fine-tuned to reflect realistic stress conditions, the pots at 25% and 50% were adjusted to 1800g and 2300g respectively and while the plant at 100% aligned with the prior weight, its moisture content was reduced by 15% to avoid root rot. These moisture percentages were maintained twice a day at 12:00 and 19:00, leading to more reliable thermal images and forming the base of our dataset.

3.2) Image acquisition and processing

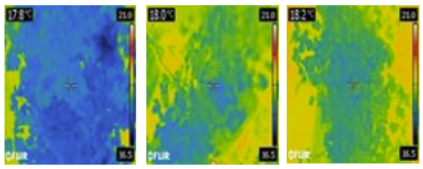
As was mentioned, the images were collected using a FLIR E60 thermal camera which captures radiometric JPGs at a resolution of 320x240 pixels. The emissivity on the image was also set at 0.98 which is appropriate for vegetation and ensures accurate surface measurements.

The temperature values existence was confirmed when using ExifTool in combination with the FlirImageExtractor library in python. Some of the images were taken with the camera on a different angle, although this switches the image size to 240x320 the thermal array was still on the original image size. It's worth noting that images taken using FLIR mobile application when transferred to the computer, they were a mere snapshot of the photo and possessed no meta data therefore there would be no temperature on the image.

The pictures valid for the final dataset were taken before the irrigation times at 11:30 and 18:30 over 5 days except for one occasion that was conducted at 13:30 due to scheduling constraints. When taking the pictures, it was aimed at 100 pictures a day and all of them were taken from above, around 40 vertical and 30 from each side on the horizontal side of the pot at a close distance leading to a balanced dataset with good angle variety yet standard and uniform for each plant. Before any photo was taken the plants were set under shade to give every leaf a more uniform temperature since then the pictures were taken under no direct sunlight radiation to avoid the temperature increasing on some leaves more than others. This leads to a data set of 1532 images (502 at no stress, 504 medium stress and 526 at high stress) with the main difference between each class being the surface temperature of the leaves.



a) no water stress; b) medium water stress; c) High water stress
Figure 1. Plants under Dry and hot weather



a) no water stress; b) medium water stress; c) High water stress
Figure 1. Plants under Humid weather

Figure 5.4. Page 3 of the article

If we compare the images on figure 1 and figure 2, we can clearly see that under humid conditions, it is hard to find a difference between the plants at low water stress and at high water stress in contrast to hot and dry conditions, therefore only images during dry and hot weather were used.

IV. Data Preparation

Data Preparation was divided into 2 parts, one path for YOLO based object detection and another for CNN based classification using the surface temperature of the extracted images.

4.1) YOLO Dataset Preparation:

The YOLO dataset had to me manually labeled using Roboflow, where bounding boxes were drawn around the parsley plant. After the labeling process, the images were resized to 640x640 for training, since YOLO works better with squared images and then some data augmentation was applied to the dataset, all this within Roboflow, which included 90-degree clockwise and counterclockwise rotations, upside-down rotations, horizontal and vertical flips, and up to 25% random cropping. This process increased the size of the dataset from 1532 images to 3676 annotated images keeping the validation and test dataset intact and augmenting only the training dataset to 3216, enhancing the model's generalization capacity across different plant orientations and positions. The number of images after augmentation were limited by Roboflow itself.

4.2) CNN Dataset Preparation:

For the CNN classification models, the radiometric JPGs were processed using the FlirImageExtractor library in combination with exiftool for metadata extraction. The temperature embedded in the images was converted into thermal arrays using numpy.

Then, a temperature offset was applied to the arrays subtracting the temperature value recorded on the day of the image, to each value in the array, making the models more robust to different temperature since it will base itself from the difference of the ambient temperature if in a day we have 30 ºC and another we have 20 ºC, the difference between the surface of a plant at low stress on each day and the ambient temperature should be around 2 ºC.

For data augmentation it was simply used horizontal and vertical flips, this choice was made because we cannot apply color changes due to it being thermal data and rotations would distort the array, this led us to have 3675 images on the training dataset.

After that to train another CNN we built another dataset fitting the YOLO bounding boxes inside the thermal arrays removing the little background that the image had and resized them to 224x224 to use as input for the transfer learning models and to 240x320 for the normal CNN using opencv.

This will do an average of the closest pixels values to fit it on the correct size (224x224 or 320x240 depending on if its transfer learning model or custom), this way we can train our data on resized thermal values, making it more robust when predicting the YOLO bounding boxes.

V. Modeling and Evaluation

5.1) CNN Modeling

On the CNN there were models with data augmentation, models. They were trained on 2 datasets, the original dataset and another with the bounding boxes of the YOLO inside the thermal numpy array removing part of the background.

There were 3 classes: No Stress, Medium Stress and High Stress. The models trained for 50 epochs, used the adam optimizer with a learning rate of 0.001, a batch size of 32 and early stop with a patience value of 5 to monitor the validation loss and if this metric does not improve, the model will end its training and save the best weights.

The data was split as 80/20 ratio as in 80% training and 20% validation considering the class distribution, then the numpy arrays were converted to a tensor flow dataset object and the training dataset will be augmented with the previous mentioned treatment.

The custom CNN were built with 3 convolutional layers, all with kernels of 3x3, the first one had 16 filters, the second 32 and the last one 48, each with a relu activation function. It possessed max pooling layers in between each convolutional layer and ended with a flatten layer to convert multi-dimensional feature maps built into a 1-dimensional vector that is fed into the fully connected layer with 48 Neurons.

A dropout of 0.5 is applied to avoid overfitting and then its finally fed into the output layer that has a softmax activation function that is used to classify the stress level of the plant.

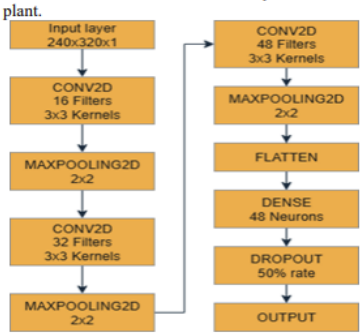


Figure 3. Custom CNN architecture

As for results, the CNN on the original dataset had an accuracy and validation accuracy of 79% meaning it did not overfit.

When data augmentation was applied there was a very small upgrade for the model allowing it to reach 80% accuracy and validation accuracy.

For transfer learning we chose the mobileNetV2 and VGG16 since these are lighter models following paper [33].

The transfer learning models had no fine-tuning and ended with a fully connected layer with 128 neurons and a dropout of 0.5.

FIGURE 5.5. Page 4 of the article

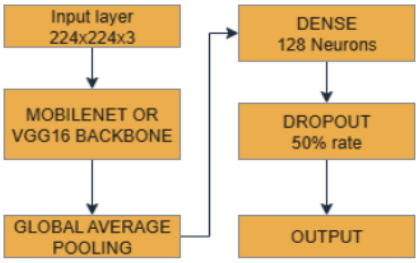


Figure 4. Transfer learning architecture

The input had to be cloned 3 times to fit the 3 channels requirements giving the array a shape of (224,224,3) and ended with a global average pooling 2d as it presented both a better performance and a lower parameter value in transfer learning.

VGG16 performed quite well with an accuracy of 99,06 % and a validation accuracy of 95.11 %. Although the results were good, it showed some overfitting and trained for a much longer time than any of the other models.

The MobileNetV2 trained on the original data was nearly perfect, with an accuracy of 99.67% and a validation accuracy of 98.70%.

This means that the model has almost perfect accuracy and generalizes very well. When compared to the VGG16, the MobileNetV2 performed much better and is a much lighter model, using a total of 2,422,339 parameters in comparison to the VGG16, 14,780,739 and much faster training.

Due to the weight of the VGG16 in terms of computational power and its performance when compared to the mobileNetV2 which is also a transfer learning model, we performed the rest of the tests (data augmentation and the background removal) only on the custom CNN and mobileNet.

With data augmentation, it dropped a bit in accuracy and validation accuracy, 98.26% and 97.72%, which is still good.

The next models were trained on the dataset which the images had their background cropped and got resized using opencv to help the model generalize the images better once they are detected by the YOLO model and have their bounding box fit into the thermal array. The architecture was kept the same as the original dataset.

For model performance, the custom model performed well once again, with an accuracy of 82.86% and a validation accuracy of 80.78% this presenting no overfitting.

The MobileNetV2 once again demonstrated superior performance when compared with the custom model but had a small decrease in quality, with an accuracy of 98.75 %, a validation accuracy of 95.44% in comparison with the results on the original dataset.

In terms of precision and recall both "High stress" and "No stress" classes presented good results, while the medium stress class was worse as we can see on the confusion matrix of the chosen classification model for example in figure 5.

Overall, the custom models had good metrics, the accuracy values round the 80% without overfitting, but transfer learning is much more powerful with the MobileNetV2 being the best model with almost perfect accuracy both on train and validation.

For the purpose of classification within our two stage object detector, although the MobileNetV2 model trained on the data with the background cropped presented worse values than the model trained on the full image, it seems more appropriate since the detector will adjust the image according to the detected bounding box and average the values, this way the model has examples to train on such data.

Table1: Evaluation metrics

Model	Accuracy	Validation Accuracy	Average Precision	Average recall
Custom	78.53%	78.83%	80%	79%
Custom w/data augmentation	80.74%	79.80%	81%	80%
MobileNetV2	99.67 %	98.70 %	99%	99%
MobileNetV2 w/dataAug	98.26%	97.72 %	98%	98%
VGG16	99.06 %	95.11 %	96%	95%
Custom No Background with data augmentation	82.86%	80.78 %	81%	81%
MobileNetV2 w/No Background	98.75%	95.44%	95%	95%
MobileNetV2 w/data augmentation and no Background	98.03%	95.77 %	96%	96%

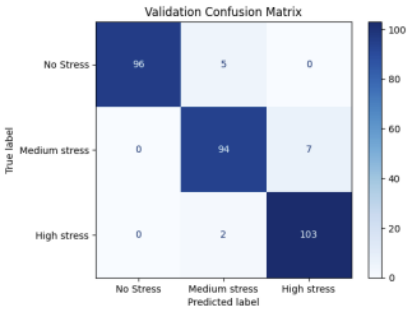


Figure 5: MobileNetV2 with data augmentation trained on the dataset with the cropped image confusion matrix

FIGURE 5.6. Page 5 of the article

5.2) Object detection model

For this paper we used YOLOv11n since it's the latest yolo version and the lightest. The dataset was split into 70%/20%/10% being 70% for training, 20% for validation and 10% for testing. Although the images on yolo were labeled with the correct classes, (No stress; Medium stress; High stress;) its main purpose will simply be to identify the plants in the images, so another model was trained with only the label "plant".

The first model (with the 3 labels) was trained for 100 epochs, a batch of 64 and a patience value of 10, the rest of the model parameters were the default YOLO values which chose the AdamW optimizer with a learning rate of 0.001. This model had a quite good performance, with a mAP50 of 0.995 and a recall of 0.997 the mAP50-95 had a metric of 0.915 averaged around all 3 classes.

The second YOLO model was trained under similar circumstances with the same parameters. Its results were similar to the last model, but had slightly worse mAP50-95, with a value of 0.875.

When testing on real data both models had similar detection performance.

VI. Conclusion

In this study a two stage approach was presented for detecting and identifying the plant thermal responses to water stress, combining object detection and temperature-based classification. In the first stage, a YOLO-based detector was employed to localize plants within thermal images efficiently with good accuracy. In the second stage, a convolutional neural network (CNN) was trained using actual thermal values to classify water stress levels, with the MobileNetV2 presenting the best values. The system performed well under controlled conditions, however, the model possesses limited generalization when applied to real-world large-scale farms, since it trained with only parsley plants under a controlled environment, but it should perform well on these specific conditions, however it serves as a foundation for future works.

ACKNOWLEDGMENTS

This work was supported by FCT/MECI through national funds and when applicable co-funded EU funds under: UID/50008: Instituto de Telecomunicações, and ManagiDiTH, 101083896 DIGITAL-2021-SKILLS-01 EU project.

REFERENCES

[1] Kurunc, A., Tezcan, N. Y., & Bimurzayev, N. (2023). Determination of Water Stress in Wheat Crops by Thermal Images Under Mediterranean Conditions. Communications in Soil Science and Plant Analysis, 54(12), 1713–1726. https://doi.org/10.1080/00103624.2023.2211089

[2] Han, M., H. Zhang, K. C. DejongeComas, and L. H. Trout. 2016. Estimating maize water stress by standard deviation of canopy temperature in thermal imagery. Agricultural Water Management 177:400–09. doi:10.1016/J.Agwat.2016.08.031.

[3] Aragão, M. F., Neto, L. G. P., De a Viana, T. V., Manzano-Juarez, J., Lacerda, C. F., Costa, J. D. N., Lima, J. S., & Azevedo, B. M. (2023). Evaluation of crop water status of melon plants in tropical semi-arid climate using thermal imaging. Revista Brasileira De Engenharia Agrícola E Ambiental, 27(6), 447–456. https://doi.org/10.1590/1807-1929/agriambi.v27n6p447-456

[4] de O. Maia Júnior, S., de Andrade, J.R., Silva, P.C. et al. Soybean Seed Priming with Brassinosteroids Mitigates the Effects of Drought Stress. J Plant Growth Regul (2024). https://doi.org/10.1007/s00344-024-11588-1

[5] Ma, S., Liu, S., Gao, Z., Wang, X., Ma, S., \& Wang, S. (2024). Water deficit Diagnosis of winter wheat based on thermal infrared imaging. Plants, 13(3), 361. https://doi.org/10.3390/plants13030361

[6] Lee, A., Kim, S., Hong, S., Han, Y., Choi, Y., Kim, M., Yun, S. K., \& Kim, G. (2019). Phenotypic analysis of fruit crops water stress using infrared thermal imaging. Journal of Biosystems Engineering, 44(2), 87–94. https://doi.org/10.1007/s42853-019-00020-2

[7] De Almeida, A. M., Coelho, R. D., Da Silva Barros, T. H., De Oliveira Costa, J., Quiloango-Chimarro, C. A., Moreno-Pizani, M. A., \& Farias-Ramírez, A. J. (2022). Water productivity and canopy thermal response of pearl millet subjected to different irrigation levels. Agricultural Water Management, 272, 107829. https://doi.org/10.1016/j.agwat.2022.107829

[8] Duarte-Galvan, C., Romero-Troncoso, R., Torres-Pacheco, I., Guevara-Gonzalez, R., Fernandez-Jaramillo, A., Contreras-Medina, L., Carrillo-Serrano, R., \& Millan-Almaraz, J. (2014). FPGA-Based smart sensor for drought stress detection in tomato plants using novel physiological variables and discrete wavelet transform. Sensors, 14(10), 18650–18669. https://doi.org/10.3390/s141018650

[9] Belfiore, N., Vinti, R., Lovat, L., Chitarra, W., Tomasi, D., De Bei, R., Meggio, F., \& Gaiotti, F. (2019). Infrared thermography to estimate vine water status: optimizing canopy measurements and thermal indices for the varieties merlot and moscato in northern Italy. Agronomy, 9(12), 821. https://doi.org/10.3390/agronomy9120821

[10] Silva, A. D. N., Ramos, M. L. G., Ribeiro, W. Q., Junior, Da Silva, P. C., Soares, G. F., Casari, R. a. D. C. N., De Sousa, C. a. F., De Lima, C. A., Santana, C. C., Silva, A. M. M., \& Vinson, C. C. (2023). Use of thermography to evaluate alternative crops for Off-Season in the Cerrado region. Plants, 12(11), 2081. https://doi.org/10.3390/plants12112081

[11] Celiktopuz, E. (2023). Determination of drought tolerance of different strawberry genotypes. PeerJ, 11, e14972. https://doi.org/10.7717/peerj.14972

[12] Renó, V., Cardellicchio, A., Romanjenko, B. C., \& Guadagno, C. R. (2024). AI-assisted image analysis and physiological validation for progressive drought detection in a diverse panel of Gossypium hirsutum L. Frontiers in Plant Science, 14. https://doi.org/10.3389/fpls.2023.1305292

[13] Kurumayya, V. (2023). Infrared imaging indices for genotype screening in plant drought responses. Acta Physiologiae Plantarum, 45(11). https://doi.org/10.1007/s11738-023-03604-w

[14] Parihar, G., Saha, S., \& Giri, L. I. (2021). Application of infrared thermography for irrigation scheduling of horticulture plants. Smart Agricultural Technology, 1, 100021. https://doi.org/10.1016/j.atech.2021.100021

[15] Orejuela-Romero, J., Chipantiza-Masabanda, J., Carrera-Oscullo, P., \& Salguero-Cajo, A. (2023). Effect of deficit irrigation on Helianthus annuus L. plants in containers. Revista De La Facultad De Agronomía, 40(2), e234012. https://doi.org/10.47280/revfacagron(luz).v40.n2.02

[16] Takács, S., Pék, Z., Csányi, D., Daood, H. G., Szuvandzsiev, P., Palotás, G., \& Helyes, L. (2020). Influence of water stress levels on the yield and lycopene content of tomato. Water, 12(8), 2165. https://doi.org/10.3390/w12082165

[17] Gutiérrez-Gordillo, S., García-Tejero, I. F., Zuazo, V. H. D., Escalera, A. G., Gil, F. F., Amores-Agüera, J. J., Rodríguez, B. C., \& Hernández-Santana, V. (2020). Assessing the Water-Stress Baselines by Thermal Imaging for Irrigation Management in Almond Plantations

Figure 5.7. Page 6 of the article

under Water Scarcity Conditions. Water, 12(5), 1298. https://doi.org/10.3390/w12051298

[18] Liyanarachchi, R., Wijekoon, J., Premathilaka, M., \& Vidhanaarachchi, S. (2023). COVID-19 symptom identification using Deep Learning and hardware emulated systems. Engineering Applications of Artificial Intelligence, 125, 106709. https://doi.org/10.1016/j.engappai.2023.106709

[19] Thaker, K., Chennupati, S., Rawashdeh, N., \& Rawashdeh, S. A. (2023). Multispectral deep neural network fusion Method for Low-Light object detection. Journal of Imaging, 10(1), 12. https://doi.org/10.3390/jimaging10010012

[20] Stancic, I., \& Saric, E. (2024). Increasing the model classification accuracy of thermal images. 2022 7th International Conference on Smart and Sustainable Technologies (SpliTech), 1–6. https://doi.org/10.23919/splitech61897.2024.10612393

[21] Y. Huangfu, L. Campbell and S. Habibi, "Temperature Effect on Thermal Imaging and Deep Learning Detection Models," 2022 IEEE Transportation Electrification Conference \& Expo (ITEC), Anaheim, CA, USA, 2022, pp. 185-189, doi: 10.1109/ITEC53557.2022.9813980.

[22] Jiang, C., Ren, H., Ye, X., Zhu, J., Zeng, H., Nan, Y., Sun, M., Ren, X., \& Huo, H. (2022). Object detection from UAV thermal infrared images and videos using YOLO models. International Journal of Applied Earth Observation and Geoinformation, 112, 102912. https://doi.org/10.1016/j.jag.2022.102912

[23] Lv, C., Mittal, U., Madaan, V., \& Agrawal, P. (2024). Vehicle detection and classification using an ensemble of EfficientDet and YOLOv8. PeerJ Computer Science, 10, e2233. https://doi.org/10.7717/peerj-cs.2233

[24] Khosa, I., Raza, A., Anjum, M., Ahmad, W., \& Shahab, S. (2023). Automatic Diabetic foot ulcer recognition using Multi-Level Thermographic Image data. Diagnostics, 13(16), 2637. https://doi.org/10.3390/diagnostics13162637

[25] Wen, M., Li, C., Xue, Y., Xu, M., Xi, Z., \& Qiu, W. (2024). YOFIR: High precise infrared object detection algorithm based on YOLO and FasterNet. Infrared Physics \& Technology, 105627. https://doi.org/10.1016/j.infrared.2024.105627

[26] Hou, Z., Sun, Y., Guo, H., Li, J., Ma, S., \& Fan, J. (2022). M-YOLO: an object detector based on global context information for infrared images. Journal of Real-Time Image Processing, 19(6), 1009–1022. https://doi.org/10.1007/s11554-022-01242-y

[27] D. Bouallal et al., "Segmentation of plantar foot thermal images: application to diabetic foot diagnosis," 2020 International Conference on Systems, Signals and Image Processing (IWSSIP), Niteroi, Brazil, 2020, pp. 116-121, doi: 10.1109/IWSSIP48289.2020.9145167.

[28] Ibrahim, A.; Anayi, F.; Packianather, M. New Transfer Learning Approach Based on a CNN for Fault Diagnosis. Eng. Proc. 2022, 24, 16. https://doi.org/10.3390/IECMA2022-12905

[29] W. Guettala, A. Sayah, L. Kahloul and A. Tibermacine, "Real Time Human Detection by Unmanned Aerial Vehicles," 2022 International Symposium on iNnovative Informatics of Biskra (ISNIB), Biskra, Algeria, 2022, pp. 1-6, doi: 10.1109/ISNIB57382.2022.10075707.

[30] Colaco, S. J., Kim, J. H., Poulose, A., Van, Z. S., Neethirajan, S., \& Han, D. S. (2022). Pig Treatment Classification on Thermal Image Data using Deep Learning. 2022 Thirteenth International Conference on Ubiquitous and Future Networks (ICUFN), 8–11. https://doi.org/10.1109/icufn55119.2022.9829713

[31] H. Aboalia, S. Hussein and A. Mahmoud, "Infrared Multi-Object Detection Using Deep Learning," 2024 14th International Conference on Electrical Engineering (ICEENG), Cairo, Egypt, 2024, pp. 175-177, doi: 10.1109/ICEENG58856.2024.10566390.

[32] E. Sousa, K. O. S. Mota, I. P. Gomes, L. Garrote, D. F. Wolf and C. Premebida, "Late-Fusion Multimodal Human Detection Based on RGB and Thermal Images for Robotic Perception," 2023 European Conference on Mobile Robots (ECMR), Coimbra, Portugal, 2023, pp. 1-6, doi: 10.1109/ECMR59166.2023.10256301.

[33] Baran, K. (2023). Smartphone thermal imaging for stressed people classification using CNN+MobileNetV2. Procedia Computer Science, 225, 2507–2515. https://doi.org/10.1016/j.procs.2023.10.242

[34] Memari, M., Shekaramiz, M., Masoum, M. a. S., \& Seibi, A. C. (2024). Data fusion and ensemble learning for advanced anomaly detection using Multi-Spectral RGB and thermal imaging of small wind turbine blades. Energies, 17(3), 673. https://doi.org/10.3390/en17030673

[35] Q. Ashfaq and M. Usman Akram, "Convolutional Neural Network Based Thermal Image Classification," 2022 2nd International Conference on Digital Futures and Transformative Technologies (ICoDT2), Rawalpindi, Pakistan, 2022, pp. 1-6, doi: 10.1109/ICoDT255437.2022.9787443.

[36] Baghezza, R., Bouchard, K., Bouzouane, A., \& Gouin-Vallerand, C. (2021). Profile recognition for accessibility and inclusivity in smart cities using a thermal imaging sensor in an embedded system. IEEE Internet of Things Journal, 9(10), 7491–7509. https://doi.org/10.1109/jiot.2021.3127137

[37] Thukral, R., Aggarwal, A. K., Arora, A. S., Dora, T., \& Sancheti, S. (2023). Artificial intelligence-based prediction of oral mucositis in patients with head-and-neck cancer: A prospective observational study utilizing a thermographic approach. Cancer Research Statistics and Treatment, 6(2), 181–190. https://doi.org/10.4103/crst.crst_332_22$

FIGURE 5.8. Page 7 of the article