ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

# INTEGRATING PATH PLANNING AND IMAGE PROCESSING WITH UAVs FOR DISEASE DETECTION AND YIELD ESTIMATION IN INDOOR AGRICULTURE

**M.Sc. THESIS**

**Onat ERDOĞMUŞ**

**Department of Mechatronics Engineering**

**Mechatronics Engineering Programme**

**July 2024**

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

# INTEGRATING PATH PLANNING AND IMAGE PROCESSING WITH UAVs FOR DISEASE DETECTION AND YIELD ESTIMATION IN INDOOR AGRICULTURE

**M.Sc. THESIS**

**Onat ERDOĞMUŞ**
**(518211019)**

**Department of Mechatronics Engineering**

**Mechatronics Engineering Programme**

**Thesis Advisor: Prof. Dr. Erdinç ALTUĞ**

**July 2024**

ISTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

# KAPALI ALAN TARIMDA HASTALIK TESPİTİ VE VERİM TAHMİNİ İÇİN ROTA PLANLAMA VE GÖRÜNTÜ İŞLEMENİN İHA'LARLA ENTEGRE EDİLMESİ

**YÜKSEK LİSANS TEZİ**

**Onat ERDOĞMUŞ**
**(518211019)**

**Mekatronik Mühendisligi Anabilim Dalı**

**Mekatronik Mühendisligi Programı**

**Tez Danışmanı: Prof. Dr. Erdinç ALTUĞ**

**Temmuz 2024**

Onat Erdoğmuş, a M.Sc. student of İTU Graduate School student ID 518211019, successfully defended the thesis entitled "INTEGRATING PATH PLANNING AND IMAGE PROCESSING WITH UAVs FOR DISEASE DETECTION AND YIELD ESTIMATION IN INDOOR AGRICULTURE", which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

| | | |
|---|---|---|
| **Thesis Advisor :** | **Prof. Dr. Erdinç ALTUĞ**<br>İstanbul Technical University | .............................. |
| **Jury Members :** | **Assoc. Prof. Dr. Merve Acer KALAFAT**<br>Istanbul Technical University | **..............................** |
| | **Dr. Mumin Tolga EMİRLER**<br>Yıldız Technical University | .............................. |

**Date of Submission : 24 May 2024**
**Date of Defense : 18 July 2024**

***To my mother, the force behind my achievements,***

**FOREWORD**

This thesis marks a major milestone in my academic journey, reflecting the knowledge I have gained and the skills I have developed during my studies.

I am thankful to Prof. Dr. Erdinç Altuğ for his guidance and support during this project.

I want to thank my family for their constant love and support. They have always believed in me and encouraged me to do my best.

I also want to thank all my teachers who have taught me over the years. Their hard work and dedication have helped me learn and grow.

Thank you to everyone who has supported me on this journey.

July 2024 Onat Erdoğmuş

**TABLE OF CONTENTS**

**Page**

## ABBREVIATIONS

| | |
|---|---|
| **ACO** | : Ant Colony Optimization |
| **ANOVA** | : Analysis of Variance |
| **CEA** | : Controlled Environment Agriculture |
| **CNN** | : Convolutional Neural Network |
| **CPU** | : Central Process Unit |
| **DFL** | : Distribution Focal Loss |
| **FLOP** | : Floating-Point Operations Per Second |
| **ICUAS** | : International Conference on Unmanned Aircraft Systems |
| **IMU** | : Inertial Measurement Unit |
| **IoU** | : Intersection over Union |
| **mAP** | : Mean Average Precision |
| **MLP** | : Multilayer Perceptron |
| **NMS** | : Non-Maximum Suppression |
| **R-CNN** | : Region-based Convolutional Neural Networks |
| **ResNet** | : Residual Network |
| **RoI** | : Region of Interest |
| **ROS** | : Robot Operating System |
| **SITL** | : Software in the Loop |
| **SLAM** | : Simultaneous Localization and Mapping |
| **SSD** | : Single Shot MultiBox Detector |
| **SVM** | : Support Vector Machine |
| **TSP** | : Traveling Salesman Problem |
| **UAV** | : Unmanned Aerial Vehicle |
| **YOLO** | : You Only Look Once |

**SYMBOLS**

**$CO_2$** **:** Carbon Dioxide

**LIST OF TABLES**

**Page**

## LIST OF FIGURES

**Page**

**INTEGRATING PATH PLANNING AND IMAGE PROCESSING WITH UAVs FOR DISEASE DETECTION AND YIELD ESTIMATION IN INDOOR AGRICULTURE**

## SUMMARY

The integration of Unmanned Aerial Vehicles (UAVs) with Controlled Environment Agriculture (CEA) is examined in this thesis, with a particular emphasis on tackling the major obstacles associated with disease detection and yield estimation in indoor farming. In order to maximize indoor agricultural practices, the research intends to take advantage of the advanced capabilities of UAVs fitted with high-resolution sensors together with complex path planning and image processing algorithms.

One of the main components of the system is the path planning module. The main task of the path planning module is to effectively guide UAVs through the small areas of indoor farms. This entails flying in the shortest path possible, avoiding obstructions like plant beds, walls and support beams, and making sure the entire greenhouse is covered. Similar to the Traveling Salesman Problem (TSP), the problem is formulated using graph theory and the objective is to find the shortest route that visits all important points.

Numerous algorithms, including heuristic approaches like Christofides' Algorithm and Nearest Neighbor, exact methods like Branch and Bound, and metaheuristic strategies like Genetic Algorithms, Ant Colony Optimization, and Simulated Annealing, were assessed. The study came to the conclusion that it would be better to use a combination of heuristic and metaheuristic strategies rather than exact algorithms. They provided the optimum compromise between computational efficiency and solution quality, and they were implemented using Google's OR-Tools library.

The path planning module was implemented by generating grid points by using the greenhouse layout and computing distance matrices. Later, the TSP was resolved by refining the early solutions using local search metaheuristics and a variety of first solution methodologies. Path Cheapest Arc, the approach that was selected for the metaheuristic comparisons, showed a consistent rate of path creation, which qualified it for more comparisons and practical deployment.

Identifying and counting fruits in snapshots that the UAVs took was part of the yield estimation task in the image processing module. Yolov8, a single-stage detector, was chosen because of its ability to merge speed and precision, which makes it perfect for real-time applications. With a high precision and recall metrics, the YOLOv8s model was trained on a dataset of 8,479 photos that included six different fruit classes. A number of measures were used to assess the model's performance, and the results showed that it was robust and effectively learned, including the Precision-Confidence Curve, F1-Confidence Curve, Recall-Confidence Curve, and Precision-Recall Curve.

The main goal of disease detection was to categorize plant leaf images in order to recognize disease symptoms. Latest architectures with great accuracy and computational efficiency, such as YOLOv8s-cls, were selected. A dataset of 18,741

photos, containing both healthy and unhealthy apple and grape leaves, was used to train the model. Confusion matrices and training loss graphs were used to evaluate the model's performance, and they verified the model's dependability and capacity to discriminate between various disease classifications and health states.

The ROS and Gazebo platforms were used for system integration and simulation. The UAV platform included key sensors and control algorithms that were integrated with the virtual environment. It was based on the Kopterworx Eagle quadcopter. With this configuration, the control techniques may be thoroughly tested and improved without the hazards that come with actual flight operations. The ROS framework enabled smooth communication between the path planning and image processing components, facilitating modular and distributed system development. The Image Processing node provided real-time picture analysis and precise yield estimation and disease detection while the Path Planner node created effective flight pathways.

The UAV was able to function as it would have in a real greenhouse given that to the simulation setup in Gazebo, which imitated a realistic indoor agricultural environment. Throughout the interior setup, the UAV moved steadily and smoothly, accurately following the created flight routes. Real-time processing of the UAV's camera's acquired visuals translated into annotated images that validated accurate yield estimations and accurate disease symptom identification. Through these simulations, the system's capacity to identify unhealthy plants and calculate yields was verified; despite a couple of discrepancies, it demonstrated great accuracy and dependability.

The study's findings suggested that an integrated system of unmanned aerial vehicles equipped with innovative path planning and image processing algorithms could substantially improve indoor agriculture's sustainability and efficiency. The dynamic time limit function of the path planning module was essential in guaranteeing effective functioning in different greenhouse sizes. The complexity of the greenhouse arrangement and the quantity of grid points were taken into account by this function while adjusting the time limit dynamically. Through iteratively executing randomized tests for varying point counts, the function determined the point at which solution distances plateud. By minimizing needless delays for simpler layouts and giving enough computing time for complex instances, this adaptive technique allowed the system to maintain superior performance. In the meantime, the image processing module's strong performance indicators highlighted how well it worked in real-time applications. Reducing the dangers and costs associated with physical trials, scalable and cost-effective testing was made possible by the use of the ROS and Gazebo simulation platforms. Additionally, the fruit detection model, tested with real-world images, demonstrated robust performance by utilizing average color analysis to filter grape varieties and reduce false positives, even under challenging conditions. The disease classification model accurately detected and classified leaves, with expert validation recommended to confirm the results, especially under non-ideal conditions.

In conclusion, this study showed how combining UAVs with novel technologies might help indoor agriculture overcome its problems with yield estimation and disease detection. In simulated situations, the suggested system demonstrated successful outcomes, demonstrating its capacity to optimize resource allocation, enhance crop management, and guarantee a steady supply of food. Future research should concentrate on scalability and implementation in real-world scenarios, field testing in various indoor farming configurations, and investigating the integration of more sophisticated sensors and enhanced UAV flight dynamics. These developments will improve the system's overall performance in revolutionizing indoor agriculture methods as well as its resilience and adaptability.

# KAPALI ALAN TARIMDA HASTALIK TESPİTİ VE VERİM TAHMİNİ İÇİN ROTA PLANLAMA VE GÖRÜNTÜ İŞLEMENİN İHA'LARLA ENTEGRE EDİLMESİ

## ÖZET

Bu tezde, İnsansız Hava Araçları (İHA) ile kontrollü ortam tarımının entegrasyonu incelenmektedir. Özellikle, iç mekan tarımında hastalık tespiti ve verim tahmini ile ilgili büyük engellerin aşılması üzerinde durulmaktadır. Araştırma, yüksek çözünürlüklü sensörlerle donatılmış İHA'ların gelişmiş yeteneklerinden yararlanarak ve karmaşık rota planlama ve görüntü işleme algoritmalarını kullanarak iç mekan tarım uygulamalarını en üst düzeye çıkarmayı amaçlamaktadır.

Sistemin temel komponentlerinden biri olan rota planlama modulünün ana görevi, İHA'ları iç mekan çiftliklerinin dar alanlarında etkin bir şekilde yönlendirmektir. Bu, bitki yatakları, duvarlar ve destek kirişleri gibi engellerden kaçınarak mümkün olan en kısa rotada uçmayı ve seranın tamamını kapsadığından emin olmayı içerir. Bu problem, Gezgin Satıcı Problemi'ne (TSP) benzer şekilde graf teorisi kullanılarak formüle edilmiştir ve amaç, tüm önemli noktaları ziyaret eden en kısa rotayı bulmaktır.

Christofides Algoritması ve En Yakın Komşu gibi sezgisel yaklaşımlar, Dal ve Sınır gibi kesin yöntemler ve Genetik Algoritmalar, Karınca Kolonisi Optimizasyonu ve Benzetimli Tavlama gibi meta-sezgisel stratejiler dahil olmak üzere birçok algoritma değerlendirildi. Bu değerlendirmelerin çıktısına bakıldığında ve insansız hava araçlarının sahip olduğu işlemci gücü veyahut zaman sınırlamaları gibi faktörlerin de değerlendirilmesiyle çalışmada, sezgisel ve meta-sezgisel stratejilerin kombinasyonunun kesin algoritmalardan daha iyi olacağı sonucuna varıldı. Bu kombinasyon, hesaplama verimliliği ve çözüm kalitesi arasında optimum dengeyi sağlamış ve Google'ın OR-Tools kütüphanesi kullanılarak uygulandı.

Rota planlama modülü, seranın düzenini kullanarak ızgara noktaları oluşturmak ve mesafe matrislerini hesaplamak suretiyle problemin tabanını oluşturmaktadır. Daha sonra, TSP, yerel arama meta-sezgiselleri ve çeşitli ilk çözüm metodolojileri kullanılarak erken çözümlerin rafine edilmesiyle çözülmektedir. Meta-sezgisel karşılaştırmalar için seçilen yöntem olan "Path Cheapest Arc", hızlı bir ilk rota oluşturma hareketi gösterdi ve daha fazla karşılaştırma ve pratik uygulama için uygun olduğunu kanıtladı. Seçilen ilk çözüm stratejisi kullanılarak farklı nokta sayıları ve farklı mesafe matrisleriyle testler çoklandı. Problem boyutuna bağlı olarak farklı meta-sezgisel yöntemlerin davranışları değerlendirildi.

İHA'ların çektiği anlık görüntülerdeki meyvelerin tespit edilmesi ve sayılması, görüntü işleme modülünde verim tahmini görevinin bir parçasıdır. Yolov8, hız ve hassasiyeti birleştirme yeteneği sayesinde tek aşamalı dedektörler arasından seçildi ve bu kategoriye mensup diğer dedektörler gibi YOLO'nun da gerçek zamanlı uygulamalar için avantajlı bir konumda olduğu tekrar kanıtlanmış oldu. Yüksek hassasiyet ve geri çağırma metriklerine sahip YOLOv8s modeli, altı farklı meyve sınıfını içeren 8,479 fotoğraftan oluşan bir veri seti üzerinde eğitildi. Modelin performansını

değerlendirmek için bir dizi ölçüm kullanıldı ve sonuçlar, modelin dayanıklı ve etkili bir şekilde öğrenildiğini gösterdi. Bu ölçümler arasında Hassasiyet-Güven Eğrisi, F1-Güven Eğrisi, Geri Çağırma-Güven Eğrisi ve Hassasiyet-Geri Çağırma Eğrisi yer almaktadır.

Hastalık tespitinin ana amacı, hastalık belirtilerini tanımak için bitki yaprağı görüntülerini kategorize etmektir. YOLOv8s-cls gibi yüksek doğruluk ve hesaplama verimliliğine sahip en son mimariler seçildi. Modeli eğitmek için hem sağlıklı hem de sağlıksız elma ve üzüm yapraklarını içeren 18.741 fotoğraftan oluşan bir veri seti kullanıldı. Modelin performansını değerlendirmek için karmaşıklık matrisleri ve eğitim kaybı grafikleri kullanıldı ve bunlar, modelin çeşitli hastalık sınıflandırmaları ve sağlık durumları arasında ayrım yapma kapasitesini ve güvenilirliğini doğruladı.

Sistem entegrasyonu ve simülasyon için ROS ve Gazebo platformları kullanıldı. İHA platformu, sanal ortamla entegre edilmiş temel sensörler ve kontrol algoritmalarını içermektedir ve Kopterworx Eagle quadcopter baz alınarak oluşturulmuştur. Bu konfigürasyon sayesinde, gerçek uçuş operasyonlarıyla gelen tehlikeler olmadan kontrol teknikleri ayrıntılı olarak test edilip geliştirilebildi. İşlemci gücü ve benzeri teknik kısıtlamaların da bu sayede önüne geçildi.  ROS çerçevesi, rota planlama ve görüntü işleme bileşenleri arasında sorunsuz iletişim sağladı ve modüler ve dağıtılmış sistem geliştirmeyi kolaylaştırdı. Görüntü İşleme düğümü, gerçek zamanlı görüntü analizi ve hassas verim tahmini ile hastalık tespiti sağlarken, Rota Planlayıcı düğümü etkili uçuş yolları oluşturdu.

Gazebo'daki simülasyon kurulumunun gerçekçi bir iç mekan tarım ortamını taklit etmesi sayesinde, İHA gerçek bir serada çalışacağı gibi işlev gösterebildi. İç mekan kurulum boyunca İHA, oluşturulan uçuş rotalarını doğru bir şekilde takip ederek sabit ve düzgün bir şekilde hareket etti. İHA'nın kamerası tarafından alınan görüntülerin gerçek zamanlı işlenmesi, doğru verim tahminlerini ve hastalık belirtisi tespitlerini doğrulayan açıklamalı görüntülere dönüştü. Bu simülasyonlar sayesinde, sistemin sağlıksız bitkileri tanımlama ve verim hesaplama kapasitesi doğrulandı; çok az sayıda hatalı algılama sayesinde sistemin büyük oranda doğruluk ve güvenilirlik gösterdi. Bu hatalı algılamalar temelinde esnek hesaplama metotlarının optimal olmayan sonuçlar vermesinden kaynaklanmaktadır.

Çalışmanın bulguları, yenilikçi rota planlama ve görüntü işleme algoritmalarıyla donatılmış insansız hava araçlarının entegre bir sisteminin, iç mekan tarımının sürdürülebilirliğini ve verimliliğini önemli ölçüde artırabileceğini öne sürmektedir. Rota planlama modülünün dinamik zaman sınırı fonksiyonu, farklı sera boyutlarında etkili işleyişi garanti altına almak için büyük bir öneme sahiptir. Bu fonksiyon, sera düzeninin karmaşıklığını ve ızgara noktalarının sayısını hesaba katarak zaman sınırını dinamik olarak ayarlamaktadır. Değişen nokta sayıları için rastgele girdili testler tekrar tekrar yürütülerek, çözüm mesafelerinin düzlüğe ulaştığı nokta belirlenmektedir. Bu uyarlanabilir teknik, daha basit düzenler için gereksiz gecikmeleri en aza indirerek ve karmaşık durumlar için yeterli hesaplama süresi sağlayarak sistemin üstün performansını korumasını sağladı. Aynı zamanda, görüntü işleme modülünün güçlü performans göstergeleri, gerçek zamanlı uygulamalarda ne kadar iyi çalıştığını da vurgulamış oldu. Fiziksel denemelerle ilişkili tehlikeleri ve maliyetleri azaltarak, ROS ve Gazebo simülasyon platformlarının kullanımı, ölçeklenebilir ve düşük maliyetli testler yapılmasını sağladı. Gerçek dünya görüntüleriyle test edilen meyve algılama modeli, zorlu koşullar altında bile üzüm çeşitlerini filtrelemek ve hatalı pozitifleri azaltmak için ortalama renk analizini kullanarak güçlü bir performans sergiledi.

Hastalık sınıflandırma modeli, ideal olmayan koşullarda uzman doğrulaması gerektirmekle beraber, yaprakları doğru bir şekilde tespit edip sınıflandırdı.

Sonuç olarak, bu çalışma, İHA'ların yenilikçi teknolojilerle birleşmesinin, iç mekan tarımının verim tahmini ve hastalık tespiti konusundaki sorunlarını aşmasına nasıl yardımcı olabileceğini gösterdi. Önerilen sistem, simüle edilmiş durumlarda başarılı sonuçlar göstererek kaynak tahsisini optimize etme, mahsul yönetimini iyileştirme ve gıda arzını sürekli olarak sağlama alanlarında etkinliğini kanıtladı. Gelecek araştırmalar, gerçek dünya senaryolarında ölçeklenebilirliğe ve uygulamaya, çeşitli iç mekan tarım konfigürasyonlarında saha testlerine ve daha gelişmiş sensörler ile geliştirilmiş İHA uçuş dinamiklerinin entegrasyonunu araştırmaya odaklanmalıdır. Bu gelişmeler, sistemin genel performansını arttırarak iç mekan tarımda kullanılan metotları ilerletecek, sistemin dayanıklılığını ve uyarlanabilirliğini güçlendirecektir.

# 1. INTRODUCTION

Indoor agriculture, often referred to as CEA, represents a paradigm shift in how food is produced. By 2050, it is expected that there will be almost 10 billion people on the planet, which will result in an unprecedented increase in food demand. [1]. Traditional farming methods, heavily reliant on vast tracts of arable land and favorable weather conditions, are increasingly challenged by climate change, urbanization, and soil degradation [2]. In contrast, indoor agriculture offers a promising solution to these challenges by enabling year-round production in controlled environments, optimizing space, and significantly improving resource utilization [3].

One of the primary advantages of indoor agriculture is its capacity for continuous production irrespective of external weather conditions. This consistency can ensure a stable food supply, which is crucial for meeting the growing global demand. Furthermore, indoor systems can precisely control environmental factors such as light, temperature, humidity, and $CO_2$ levels, creating optimal growing conditions that enhance crop yield and quality. This level of control also minimizes the need for chemical inputs like pesticides and herbicides, promoting more sustainable farming practices [4].

In terms of resource efficiency, indoor agriculture systems, particularly those employing hydroponics or aeroponics, use significantly less water than traditional farming. Hydroponic systems can reduce water usage by up to 90% compared to conventional soil-based farming [5]. This efficiency is particularly vital in regions facing water scarcity, positioning indoor agriculture as a critical component of sustainable food production strategies.

Despite these advantages, indoor agriculture faces significant challenges, notably in the areas of disease detection and yield estimation. Diseases can spread rapidly in the enclosed environments typical of indoor farms [6], potentially devastating entire crops if not detected and managed promptly. Similarly, accurately estimating crop yield is essential for planning and market supply, requiring accurate and consistent monitoring to manage. These challenges directly impact crop quality, productivity, and the overall profitability of indoor farming operations.

Traditional methods for disease detection and yield estimation in indoor agriculture primarily rely on manual inspection or fixed sensor networks. Manual inspection is labor-intensive, time-consuming, and prone to human error, while fixed sensors often lack the flexibility and coverage needed to monitor large or complex indoor farming setups effectively. As a result, there is a pressing need for more effective and efficient solutions to these challenges.

## 1.1 Literature Review

UAVs have great potential in precision agriculture by offering effective solutions for tasks such as mapping, spraying, crop monitoring, irrigation, weed detection, and remote sensing. In agricultural mapping, UAVs provide detailed imagery that supports various analytical tasks. For instance, Zhang et al. developed a deep learning method for classifying plants from UAV images, demonstrating superior performance in vegetation mapping [7]. Similarly, Schiefer et al. used UAVs with CNNs to effectively map tree species in forest environments, showcasing the utility of UAVs in complex vegetation mapping [8]. UAVs have improved the precision of agricultural spraying. For example, Faiçal et al. implemented a system that uses UAVs for precise pesticide application, integrating route-planning algorithms and sensor networks to enhance accuracy and reduce environmental impact [9]. Meng et al. evaluated UAV operational parameters, demonstrating improvements in droplet coverage for trees [10].

UAVs equipped with multispectral cameras enhance crop monitoring by providing critical data on crop health and yield. Fu et al. employed various machine learning models to estimate wheat yield from UAV imagery, showing improved prediction accuracy [11]. Cao et al.  used UAVs to calculate vegetation indices, effectively estimating the fresh weight of sugar beet crops [12]. In irrigation management, UAVs, combined with machine learning models, have introduced innovative ways for assessing plant water stress. Romero et al. developed multiple models for optimal irrigation using UAV data, offering a new tool for precise irrigation management [13]. For weed detection, UAVs enable the development of high-accuracy classification systems. dos Santos Ferreira et al. used UAV images and convolutional neural networks to classify various vegetation types in soybean fields, achieving high accuracies [14].

UAVs reduce the need for extensive ground-based work, offering a cost-effective solution for crop monitoring. Wan et al. used UAVs with multispectral cameras to predict rice grain yield, demonstrating the efficiency of UAVs in providing non-destructive crop assessments [15]. Tokekar et al. demonstrated the importance of UAV path planning in optimizing fertilizer application, reducing operational time and improving efficiency [16]. Johansen et al. achieved high-accuracy crop mapping using UAV images and machine learning classifiers, highlighting the precision of UAVs in agriculture [17].

The potential of unmanned aerial vehicles (UAVs) for disease detection in agriculture has been greatly expanded by recent developments in deep learning. Numerous studies have shown how effective these technologies are for a variety of crops and illnesses. For example, Mohanty et al. [18] achieved high accuracy rates in the detection of various diseases in crops like apple, blueberry, cherry, and corn by using deep learning architectures like AlexNet and GoogLeNet. Support Vector Machine (SVM) based techniques have been used in citrus farming by Suproteem et al. and Garcia-Ruiz et al. to identify Huanglongbing-infected crops [19, 20]. Research using cutting-edge methods such as SegNet by Kerkech et al. [21], Convolutional Neural Networks (CNNs) by Kerkech et al. [22], and ANOVA by Di Gennaro et al. [23] has made similar progress in the detection of grape disease. These technologies have also proven beneficial for tomato disease detection; Zhang et al. [24] employed a variety of deep learning models, including AlexNet, GoogLeNet, and ResNet; Abdulridha et al. [25]

applied Multilayer Perceptrons (MLPs). These varied uses highlight the adaptability and promise of UAV-based disease detection systems in various agricultural settings. The fact that businesses like Tevel Tech are utilizing these technologies to create autonomous harvesting robots only serves to highlight how useful they are for agricultural operations [26]. Even though the majority of these studies concentrate on outdoor applications, they offer a solid basis for tailoring comparable strategies to the particular difficulties associated with indoor agriculture.

The literature shows while UAVs are widely used in outdoor agriculture, their applications in indoor farming is by no means exhaustive. To name a couple examples, Roldan et al. explored UAVs with sensors for measuring environmental parameters in greenhouses [27]. Simon et al. used UAV positioning in greenhouses using fixed nodes [28]. The limited research on UAV applications in indoor farming presents a significant opportunity for innovation. Integrating more UAV technology into controlled environment agriculture will help to improve disease detection and yield estimation, enhancing the efficiency and sustainability of indoor farming operations.

## 1.2 Purpose of Thesis

The advancing field of indoor agriculture, while offering great benefits in terms of sustainability and efficiency, faces various challenges, notably in disease detection and yield estimation. This thesis aims to harness the advanced capabilities of UAVs equipped with high-resolution sensors to address these issues. By integrating path planning and image processing technologies, UAVs can be optimized for the demands of CEA. The main goal is to develop a modular and adaptable system that indoor farmers can easily integrate into their existing operations. This system combines autonomous navigation capabilities, which are crucial for the intricate environment of indoor farms, with robust image processing tools that leverage deep learning for precise disease detection and yield estimation. By focusing on the autonomous operation of UAVs within indoor settings, this research aims to utilize these vehicles to navigate and function without extensive human intervention, thus streamlining the monitoring and management processes.

To achieve these objectives, this thesis will develop an efficient path planning system specifically tailored for the confined spaces of indoor farms. Additionally, an image recognition system using state-of-the-art deep learning algorithms will be implemented to accurately identify signs of disease and estimate crop yield. Both systems will be integrated and tested within a simulated environment using ROS and Gazebo simulation platform. The choice of ROS/Gazebo over real-life testing has several technical and practical considerations. Since the computational demand of the deep learning system may be too high, the system must be perfected using more robust hardware, rather than the hardware on typically available on small, commercially feasible drones. The ROS/Gazebo environment also facilitates scalable testing, reducing the risk of costly physical trials and allowing the examination of system effectiveness.

# 2. METHODOLOGY

## 2.1 System Overview

The proposed system integrates path planning and image processing techniques with UAVs to allow disease detection and yield estimation in indoor agriculture. This high-level architecture consists of three main components: the Path Planning Module, the Image Processing Module, and various UAV platform integrated through the ROS.

The Path Planning Module includes both a path planner and a trajectory planner. The path planner is responsible for generating efficient flight paths that ensure comprehensive coverage of the greenhouse area. It takes into account the geometric constraints of the indoor environment and the operational parameters of the UAV. The trajectory planner further refines these paths, ensuring smooth and feasible movements that the UAV can follow.

The Image Processing Module is the core analytical component of the system. Utilizing advanced computer vision techniques, it processes the visual data captured by the UAV's camera. This module performs tasks such as disease detection and yield estimation by analyzing images for symptoms of plant diseases and assessing the number of crops.

The UAV platform encompasses the virtual drone equipped with essential sensors for navigation and data collection, such as a camera, an IMU and odometry plugins. While specific components for SLAM and other sensor-based functionalities are crucial for the UAV’s operation, their developments are beyond the scope of this study.

Integration and control within this system is managed using ROS, which facilitates communication and coordination between the different modules. ROS handles the data flow, command execution, and real-time adjustments needed for the UAV's operations, ensuring that path planning and image processing are synchronized effectively.

## 2.2 Path Planning

### 2.2.1 Problem formulation

The primary challenge is to cover the entire area of the greenhouse while minimizing flight time and ensuring that the UAV can navigate the constrained indoor environment. The geometric constraints of the greenhouse, including plant beds, walls, and support beams, require careful path planning to avoid collisions. Additionally, the UAV must cover the entire area to ensure complete data collection. The UAV's operational parameters, such as flight time, battery life, and payload capacity, further complicate the optimization of its path.

Mathematically, the path planning problem can be formulated using graph theory. Consider a set of points P = {$p_1$, $p_2$, ..., $p_n$} representing the key locations within the greenhouse. The goal is to determine the shortest path that makes visits to each point and returns to the starting point., similar to the TSP. Let $d_{ij}$ denote the distance between point $p_i$ and point $p_j$, and define $x_{ij}$ as a binary variable where $x_{ij} = 1$ if the path includes traveling directly from point $p_i$ to point $p_j$, and $x_{ij} = 0$ otherwise. The problem can be formulated as follows:

$$Minimize \quad \sum_{i=1}^{n}\sum_{j=1}^{n} d_{ij}x_{ij} \tag{2.1}$$

Subject to:

$$\sum_{j=1}^{n} x_{ij} = 1 \quad \forall\, i \in \{1, \dots, n\} \tag{2.2}$$

$$\sum_{i=1}^{n} x_{ij} = 1 \quad \forall\, j \in \{1, \dots, n\} \tag{2.3}$$

$$xij \in \{0, 1\} \quad \forall\, i, j \in \{1, \dots, n\} \tag{2.4}$$

These constraints ensure that each point is visited exactly once and that the UAV returns to the starting point. The objectives of the path planning module are to minimize flight time, ensure complete coverage, and consider geometric constraints. By solving the TSP, the UAV follows the shortest possible path, reducing overall flight time and conserving battery life, while ensuring comprehensive coverage of the greenhouse.

### 2.2.2 Comparison of different TSP approaches

TSP is a classic optimization problem where the objective is to determine the shortest possible route that visits each point exactly once and returns to the starting point. It is an NP-hard problem in combinatorial optimization, important in theoretical computer science and operations research. Numerous approaches have been developed to solve the TSP, each with distinct advantages and limitations, particularly in terms of optimality, computational complexity, and scalability.

Exact algorithms, such as Branch and Bound or Dynamic Programming, offer guaranteed optimal solutions but are computationally expensive. The Branch and Bound approach systematically explores all possible routes to find the optimal solution, using bounds to eliminate suboptimal routes early [29]. This method ensures optimality but suffers from exponential time complexity, making it impractical for large instances. Dynamic Programming, specifically the Held-Karp algorithm, breaks the TSP into smaller subproblems and solves them recursively [30]. While it also guarantees an optimal solution and is more efficient than a brute force approach, it still has exponential time and space complexity, limiting its applicability to small to medium-sized problems.

Heuristic algorithms provide faster, though often suboptimal, solutions. The Nearest Neighbor heuristic starts from an initial point and repeatedly visits the nearest unvisited point until all points are visited. This method is computationally efficient and easy to implement but tends to yield suboptimal routes [31]. Christofides' algorithm, designed for metric TSPs, constructs a solution within a factor of 1.5 of the optimal solution. It involves creating a minimum spanning tree, finding a minimum-weight matching, and combining these to form a tour [32]. While more complex than the Nearest Neighbor heuristic, Christofides' algorithm offers better guarantees on solution quality.

Metaheuristic algorithms, such as Genetic Algorithms, Ant Colony Optimization, and Simulated Annealing, provide flexible and adaptable solutions suitable for large instances of TSP. Genetic Algorithms draw inspiration from natural selection, employing operations like selection, crossover, and mutation to evolve a population of solutions over generations [33]. They can handle large problem instances effectively but require careful parameter tuning and do not guarantee optimality. Ant Colony Optimization simulates the foraging behavior of ants, where ants deposit pheromones on paths they travel, with shorter paths receiving more pheromone concentration, this guides subsequent ants to follow these paths [34]. This makes ACO effective for large-scale TSPs, although it can be computationally intensive and slow to converge. Simulated Annealing mimics the process of annealing in metallurgy, exploring the solution space by accepting not only improvements but also worse solutions with a decreasing probability to escape local optima. This method is simple to implement and effective for various optimization problems but requires a carefully designed cooling schedule [35].

The chosen method for solving the TSP in this study leverages a combination of heuristic and metaheuristic strategies. The approach involves setting up a data model with the distance matrix and then utilizing first solution strategies and local search metaheuristics with the help of Googles’ OR-Tools library [36]. The following descriptions and methodologies in this thesis are in line with the OR-Tools documentation, unless otherwise stated. First solution strategies include methods such as Path Cheapest Arc, which finds the cheapest route from each point, and Global Cheapest Arc, which finds the cheapest route globally. Local search metaheuristics such as Guided Local Search and Simulated Annealing are used to refine the initial solution by exploring the neighborhood of current solutions to find improvements. These strategies provide a much-needed balance between solution quality and computational efficiency, making them suitable for large and complex TSP instances in the indoor agriculture scenarios.

A summary of the TSP approaches regarding their optimality, computational complexity, scalability, strengths and weaknesses is given in Table 2.1. This comparison highlights the trade-offs between different TSP solving approaches. Exact algorithms provide guaranteed optimal solutions but are limited by computational constraints, while heuristic and metaheuristic algorithms offer greater scalability and flexibility at the cost of guaranteed optimality. The chosen combination of heuristic and metaheuristic strategies balances these trade-offs, offering an efficient and effective solution for the TSP in our indoor agriculture application.

**Table 1.1:** Summary of Different TSP Approaches

| Approach | Optimality | Computational Complexity | Scalability | Strengths | Weaknesses |
|---|---|---|---|---|---|
| Branch and Bound | Guaranteed | Exponential | Poor | Finds optimal solution | Impractical for large instances |
| Dynamic Programming | Guaranteed | Exponential | Poor | More efficient than brute force | Limited to small-medium problems |
| Nearest Neighbor | Suboptimal | Polynomial | Good | Simple and fast | Often yields poor quality solutions |
| Christofides' Algorithm | Within 1.5 of optimal | Polynomial | Fair | Better quality than nearest neighbor | More complex to implement |
| Genetic Algorithms | Typically high-quality | Polynomial | Very good | Flexible, adaptable | Requires parameter tuning |
| Ant Colony Optimization | Typically high-quality | Polynomial | Very good | Effective for large problems | Computationally intensive |
| Simulated Annealing | Typically high-quality | Polynomial | Good | Simple to implement | Sensitive to cooling schedule |

### 2.2.3 Implementation details

The implementation of the path planning module integrates several critical components and processes to guarantee accurate UAV navigation. The system begins by creating grid points based on the input coordinates and plant bed locations. Each grid point is assigned a unique identifier, accounting for different UAV orientations. Afterwards the distance matrix calculation is done, it is a key step where the system computes distances between grid points, considering walls and other obstacles. If a wall is detected between two points, the calculated distance is adjusted according to the shortest path passing around this wall.

The data model prepared for OR-Tools includes elements such as the distance matrix, starting point, and the end point. The function to solve the TSP sets up the routing index manager, which manages the mapping between location indices and node indices in the underlying network model. Defines distance callback, the function to return the distance between any two locations using the distance matrix. This is used by the solver to evaluate route costs. Later it configures search parameters, including various first solution strategies and local search metaheuristics. A detailed architecture of the Path Planning Module is shown in Figure 2.1.

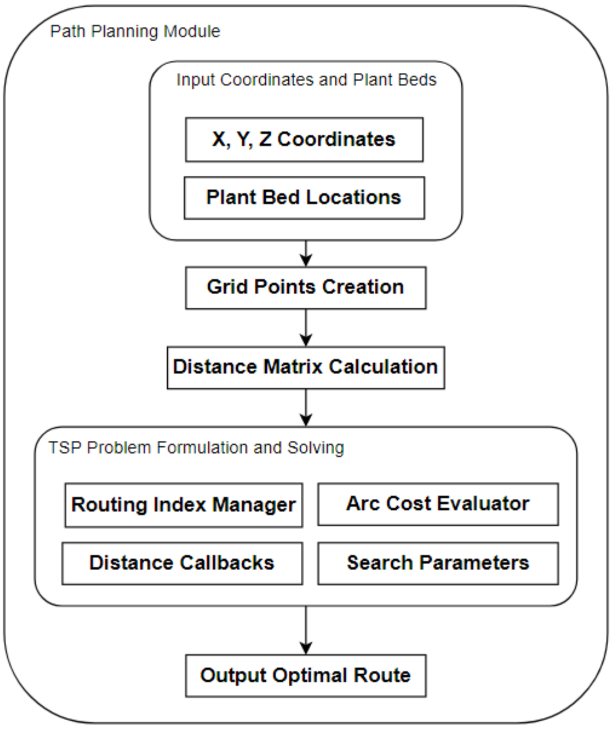


**Figure 2.1 :** Path Planning Module Architecture.

First solution strategies provide initial feasible solutions to the TSP. To mention a few, "Path Cheapest Arc" starts from a route node and iteratively connects it to the node that produces the cheapest segment, extending the route based on the last node added. "Global Cheapest Arc" finds the globally cheapest route segment iteratively. "Savings" algorithm, based on Clarke and Wright's method [37], constructs routes by merging savings in route costs. "Local Cheapest Insertion" builds a solution iteratively by inserting each node at its cheapest position, based on arc cost, differing from “Parallel Insertion” by considering nodes in their order of creation.

Local search metaheuristics refine these initial solutions by exploring their neighborhoods for improvements. "Guided Local Search" escapes local minima by penalizing certain elements in the search, guiding the algorithm to explore new areas [38]. "Simulated Annealing" mimics the annealing process in metallurgy, allowing occasional acceptance of worse solutions to escape local optima, which helps in exploring a broader solution space [39]. "Tabu Search" uses memory structures that describe the visited solutions, preventing the algorithm from revisiting them and thus escaping local minima [40]. "Generic Tabu Search" extends this by focusing on the objective value of solutions. "Greedy Descent" accepts only cost-reducing moves until a local minimum is reached, providing a straightforward yet effective approach for local optimization.

The proposed path planning approach is highly scalable and flexible, capable of adapting to different greenhouse sizes and layouts. By inputting necessary parameters such as wall-border coordinates and plant bed locations, the system automatically creates a grid-map of the greenhouse and a distance matrix based on it. This eliminates the need for manual adjustments, making the system easily adaptable to various indoor agricultural environments.

## 2.3 Image Processing

### 2.3.1 Yield estimation

#### 2.3.1.1 Problem definition and requirements

Yield estimation in indoor agriculture involves determining the quantity and size of crops grown in a greenhouse or other controlled environment. This task is crucial for optimizing resource allocation and planning market supply. Accurate and real-time yield estimation allows farmers to allocate resources like water and nutrients more effectively, plan their supply chains to meet market demand without surplus or shortage, and make informed decisions on planting schedules and labor management. The requirements for such a system include high accuracy, real-time processing capabilities, scalability to various indoor farming setups, robustness under different environmental conditions, and ease of integration with existing agricultural management systems and UAV platforms.

#### 2.3.1.2 Object detection approaches

In real-time UAV applications for yield estimation, object detection algorithms must balance speed, accuracy, and model size. Single-stage detectors are particularly suitable for these applications due to their ability to process images in a single pass, providing faster inference times compared to two-stage detectors. Single-stage detectors directly predict object classes and locations, making them efficient for real-time processing on UAVs. The model's design allows for quick and accurate detection, which is essential for the dynamic environment of indoor agriculture.

Two-stage detectors, such as Region-based Convolutional Neural Networks, including Fast R-CNN and Faster R-CNN, first propose regions of interest (RoIs) and then classify these regions [41]. While these approaches can achieve high accuracy, they can also be computationally intensive and slower, making them less suitable for real-time applications where speed is critical. Single-stage detectors, like Single Shot MultiBox Detector (SSD), bypass the region proposal step, directly predicting object classes and locations [42]. This streamlined process results in faster inference times, making single-stage detectors more appropriate for real-time UAV applications.

Anchor-based approaches use predefined anchor boxes to predict bounding boxes around objects, facilitating a robust and efficient detection process. Anchors provide initial guesses for object locations, which the network then refines. This method simplifies the detection pipeline and reduces computational complexity, making it well-suited for the limited hardware capabilities of UAVs. On the other hand, anchor-free methods, such as CenterNet, predict key points directly without predefined anchors [43]. Which can simplify implementation but often require more sophisticated network designs to achieve comparable accuracy.

Technical concepts such as Intersection over Union (IoU) and Non-Maximum Suppression (NMS) play crucial roles in object detection performance. IoU measures the overlap between the predicted bounding box and the ground truth, with values ranging from 0 to 1, where higher values indicate better performance. NMS is a technique used to eliminate redundant bounding boxes that predict the same object, ensuring that only the bounding box with the highest confidence score is retained while others that overlap significantly are suppressed. By leveraging these techniques, efficient and reliable object detection is achieved.

Given these considerations, the choice of object detection method must balance speed and accuracy to ensure real-time yield estimation. Single-stage, anchor-based methods can provide a balanced solution for UAV-based applications in indoor agriculture due to their speed, robustness, and efficiency.

#### 2.3.1.3 Model and dataset

For the task of yield estimation in indoor agriculture using UAVs, selecting an efficient and accurate object detection model is vital. Among the various object detection algorithms, the single-stage, anchor-based detector YOLO (You Only Look Once) stands out due to its balance of speed and accuracy, making it ideal for real-time applications on UAVs. The YOLOv8 model family is chosen for this study based on its excellent performance metrics [44]. All the following data in this thesis related to YOLOv8 models and methodologies are taken from the official documentation, unless otherwise stated. Table 2.2 provides the details of the YOLOv8 detection models performance metrics from their training on the COCO dataset.

**Table 1.2:** YOLOv8 COCO dataset detection trainining results

| Model | Size (pixels) | mAP (val 50-95) | Speed (CPU ONNX) (ms) | Speed (A100 TensorRT) (ms) | Params (M) | FLOPs (B) |
|---|---|---|---|---|---|---|
| YOLOv8n | 640 | 37.3 | 80.4 | 0.99 | 3.2 | 8.7 |
| YOLOv8s | 640 | 44.9 | 128.4 | 1.20 | 11.2 | 28.6 |
| YOLOv8m | 640 | 50.2 | 234.7 | 1.83 | 25.9 | 78.9 |
| YOLOv8l | 640 | 52.9 | 375.2 | 2.39 | 43.7 | 165.2 |
| YOLOv8x | 640 | 53.9 | 479.1 | 3.53 | 68.2 | 257.8 |

The metrics demonstrate that YOLOv8s provides a mean Average Precision (mAP) of 44.9% with 11.2 million parameters and 28.6 billion floating-point operations per second (FLOPs), while maintaining an inference speed of 128.4 milliseconds on a CPU using ONNX. In comparison, the YOLOv8n model, which is smaller and faster but its mAP of 37.3% may not provide sufficient accuracy for reliable yield estimation. While YOLOv8m, YOLOv8l, and YOLOv8x models offer higher mAP values of 50.2%, 52.9%, and 53.9% respectively, they come at the cost of increased parameters, FLOPs, and notably slower inference speeds.

Given these metrics, YOLOv8s strikes an optimal balance between computational efficiency and detection accuracy, making it the most suitable choice for yield estimation in UAV applications. The YOLOv8s model is particularly well-suited for real-time applications on UAVs due to its relatively high mAP and manageable computational load, ensuring that the UAV can process images swiftly and accurately while navigating. This balance is crucial for practical deployment, where the UAV needs to perform real-time detection without being hindered by excessive computational demands, which could affect flight stability and operational safety.

YOLOv8s operates on a single-stage detection mechanism consisting of 225 layers, comprising key components such as the input layer, feature extraction layers (including convolutional layers, residual blocks, and C2f blocks), and the detection head responsible for predicting bounding boxes and class probabilities. This streamlined architecture ensures efficient handling of multiple scales and sizes of objects.

The "Fruit Detection Dataset" by Multilabel Fruits Detection is employed to train and evaluate the YOLOv8s model [45]. This dataset includes 8,479 images covering six different fruit classes: Apple, Grapes, Pineapple, Orange, Banana, and Watermelon. Sample images are shown in Figure 2.2. Each image is annotated with bounding boxes, providing the ground truth for training the object detection model. The dataset is rich and diverse, capturing different orientations, lighting conditions, and occlusions. This way a comprehensive training ground is built, which is essential for training a robust detection model.

**Figure 2.2 :** Fruit Detection Dataset sample images

### 2.3.2 Disease detection (classification)

#### 2.3.2.1 Problem definition and requirements

Disease detection in agriculture is essential for maintaining crop health and maximizing yield. The primary goal is to identify disease symptoms early to prevent their spread and minimize impact. Early and accurate disease detection facilitates timely interventions, reducing crop losses. Therefore, robust and efficient image classification techniques capable of real-time operation are necessary.

An effective disease detection system must meet several key requirements. High accuracy is crucial, as the model must accurately differentiate between healthy and diseased plants, as well as among different types of diseases. Real-time processing is essential given the dynamic nature of UAV operations, necessitating rapid image analysis on-the-fly. Robustness is necessary to handle variations in lighting and plant orientation, assuring reliable performance under diverse environmental conditions. Finally, ease of integration is important so the model can seamlessly work with existing agricultural management systems and UAV platforms, facilitating smooth adoption and operation.

#### 2.3.2.2 Image classification approaches

In the context of real-time UAV applications for disease detection, image classification models need to balance speed, accuracy, and computational efficiency. Traditional convolutional neural networks (CNNs) like AlexNet and VGGNet have laid the groundwork for deep learning in image classification. AlexNet, introduced in 2012, demonstrated substantial performance improvements over traditional methods with its five convolutional layers followed by three fully connected layers [46]. However, it is relatively large and computationally demanding. VGGNet, developed by the Visual Geometry Group at Oxford, uses a deeper architecture with smaller convolutional filters (3x3). While it achieves high accuracy, the increased depth and number of parameters lead to higher computational costs and memory usage, making it less suitable for real-time UAV applications [47].

Modern architectures address these limitations while providing improved performance. GoogLeNet employs the Inception module, which uses filters of multiple sizes within the same layer to capture different levels of detail. This architecture reduces the number of parameters compared to traditional CNNs, addressing computational efficiency while maintaining high accuracy [48]. Additionally, GoogLeNet introduces auxiliary classifiers to combat the vanishing gradient problem, enhancing its robustness. ResNet (Residual Network) introduces residual connections or "skip connections" that allow gradients to flow more easily through the network during training. This design addresses the vanishing gradient problem effectively, enabling the training of very deep networks with significantly improved performance [49]. ResNet architectures, ranging from ResNet-18 to ResNet-152, vary in depth and complexity, offering superior depth and accuracy with manageable computational complexity.

Given the need for real-time processing and accuracy, modern architectures are the clear choices. They provide the computational efficiency and robustness necessary for UAV-based disease detection in indoor agriculture. This kind of balance of speed, accuracy, and efficiency is needed for integration with UAV systems.

#### 2.3.2.3. Model and dataset

For this task classification task, a model with a modern architecture, the YOLOv8s-cls model was selected due to its performance metrics. This model provides a good trade-off between accuracy and speed, which essential for the task at hand. Table 2.3 provides the details of the YOLOv8 classification models performance metrics from its' training on the ImageNet dataset.

**Table 1.3:** YOLOv8 ImageNet dataset classification trainining results

| Model | Size (pixels) | Top-1 Accuracy | Top-5 Accuracy | Speed (CPU ONNX, ms) | Speed (A100 TensorRT, ms) | Parameters (M) | FLOPs (B) |
|---|---|---|---|---|---|---|---|
| YOLOv8n-cls | 224 | 69.0% | 88.3% | 12.9 | 0.31 | 2.7 | 4.3 |
| YOLOv8s-cls | 224 | 73.8% | 91.7% | 23.4 | 0.35 | 6.4 | 13.5 |
| YOLOv8m-cls | 224 | 76.8% | 93.5% | 85.4 | 0.62 | 17.0 | 42.7 |
| YOLOv8l-cls | 224 | 76.8% | 93.5% | 163.0 | 0.87 | 37.5 | 99.7 |
| YOLOv8x-cls | 224 | 79.0% | 94.6% | 232.0 | 1.01 | 57.4 | 154.8 |

The YOLOv8s-cls model with a top-1 accuracy of 73.8%, 6.4 million parameters, and 13.5 billion FLOPs, it maintains an inference speed of 23.4 milliseconds on a CPU using ONNX, demonstrating its lightweight nature and suitability for UAVs.

The architecture of the YOLOv8s-cls model includes 99 layers, featuring key components such as the input layer, feature extraction layers (convolutional layers, residual blocks, and C2f blocks), and the detection head for predicting class probabilities. This architecture ensures the model can handle multiple categories and sizes of objects efficiently, making it ideal for classifying plant diseases in a UAV's real-time operational context.

In addition to the YOLOv8s-cls model, the pre-trained FODUU AI YOLOv8s leaf detection model was also utilized [50]. This model demonstrated a precision (mAP@0.5) of 0.946 on the object detection task, showcasing its effective object detection capability. This high precision underscores the model's suitability for accurate leaf detection in agricultural applications. The classification model runs on the bounding box which is detected by this leaf detection model.

The dataset employed for training and evaluating the YOLOv8s-cls model consists of 18,741 images, selected from a larger dataset (New Plant Diseases Dataset) of 87,000 images [51]. The focus of this study is on two types of fruits: apples and grapes. The dataset includes images of the following leaf types: Apple Scab, Apple Black Rot, Cedar Apple Rust, Healthy Apple, Grape Black Rot, Grape Esca (Black Measles), Grape Leaf Blight (Isariopsis Leaf Spot), and Healthy Grapes. These images are categorized for classification, providing a suitable training ground for developing a robust classification model.

## 2.4 System Integration and Simulation

### 2.4.1 UAV platform

The UAV platform utilized in this study is modeled after the Kopterworx Eagle, a versatile and reliable quadcopter. This UAV features a rigid body with four arms and propellers, simulating real-world propeller dynamics through a dedicated rotor simulation package. The UAV is equipped with an IMU and odometry plugins, which provide essential attitude and position data crucial for navigation and control. The model used in Gazebo is shown in Figure 2.3.

To achieve realistic behavior and control testing, the UAV employs a software-in-the-loop (SITL) control paradigm. The SITL approach integrates the UAV's control algorithms with a simulated environment, allowing widespread testing and refinement of control strategies without the risks associated with real-world flights. This paradigm is implemented using a complete control stack, which includes flight control software, sensor data processing modules, and communication interfaces. The control stack, along with the UAV platform and base Gazebo models, are adapted from the ICUAS24' UAV Competition GitHub repository [52].

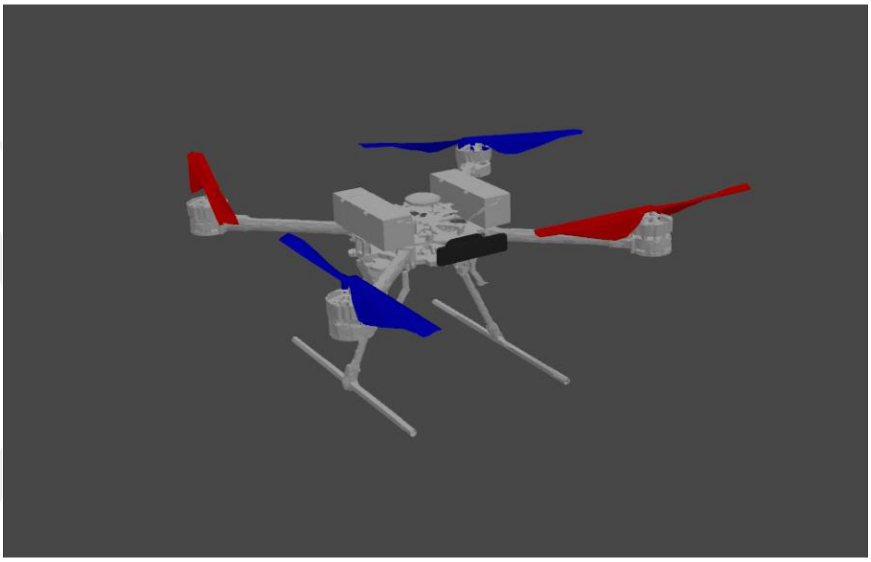

**Figure 2.3 :** UAV model

The control stack enables realistic UAV behavior in the simulation environment by closely mimicking the UAV's response to control inputs and environmental factors. Additionally, the use of Docker to package all necessary software components into a complete simulation stack ensures a consistent and reproducible simulation environment.

### 2.4.2 Communication and the ROS architecture

The communication and control architecture of the proposed UAV system is based on the Robot Operating System (ROS), a flexible framework for writing robot software [53]. ROS provides the necessary tools and libraries to develop modular and distributed systems, which is crucial for integrating various software modules and hardware components in this project. This architecture facilitates efficient communication and synchronization between the path planning and image processing modules, ensuring seamless operation of the UAV.

In this system, ROS enables modularity by allowing different functionalities to be developed as independent nodes. Each node performs a specific task, such as path planning, trajectory planning or image processing. These nodes communicate with each other using ROS topics, services, and messages, creating a network of interconnected components. This approach provides code reusability and scalability, making it easier to debug and extend the system.

The Path Planner node generates efficient flight paths considering the geometric constraints of the indoor environment and the UAV's operational parameters. It publishes the generated paths to the ‘path’ topic. The Trajectory Planner node subscribes to this topic, refines the paths into smooth and feasible trajectory points, and then publishes the trajectory points to the ‘trajectory’ topic. The UAV control stack subscribes to this topic to receive navigation commands.

The Image Processing node is responsible for analyzing the visual data captured by the UAV's camera. It subscribes to the camera feed through the ‘camera’ topic, which outputs the raw image data taken from the camera sensor. Upon receiving a trigger from the 'image_signal' service, it processes the images to detect plant diseases and estimate yields. Once processing is complete, it sends a signal back to the Trajectory Planner node via the 'image_signal' service, indicating that the UAV can continue its inspection tasks. Annotated images are published to the 'annotated_image' topic for visualization.

Additionally, the Path Planner and Trajectory Planner nodes subscribe to the 'odometry' topic to receive real-time positional data, ensuring accurate path planning and trajectory tracking. This setup ensures that the UAV navigates efficiently through the greenhouse, capturing high-quality images for processing.

### 2.4.3 Simulation setup in ROS/Gazebo

The simulation setup in ROS/Gazebo is essential for validating the integrated path planning and image processing system for UAVs in indoor agriculture. Gazebo, a powerful robot simulation tool, creates a realistic environment where UAVs can operate as they would in a physical greenhouse, allowing extensive testing without the risks and costs of real-world flights [54].

The simulated environment closely mimics an indoor agricultural setup, featuring shelves with plant beds and images of tree branches to replicate crops. This setup allows effective testing of image processing algorithms for disease detection and yield estimation. Gazebo’s physics engine simulates the UAV's motion and dynamics, including the effects of gravity and collisions. The UAV's flight dynamics were accurately modeled to provide realistic control responses and maneuverability. Figure 2.4. shows the model used for plant bed racks and Figure 2.5. shows the generated grid points, to be used in TSP, based on that rack geometry.

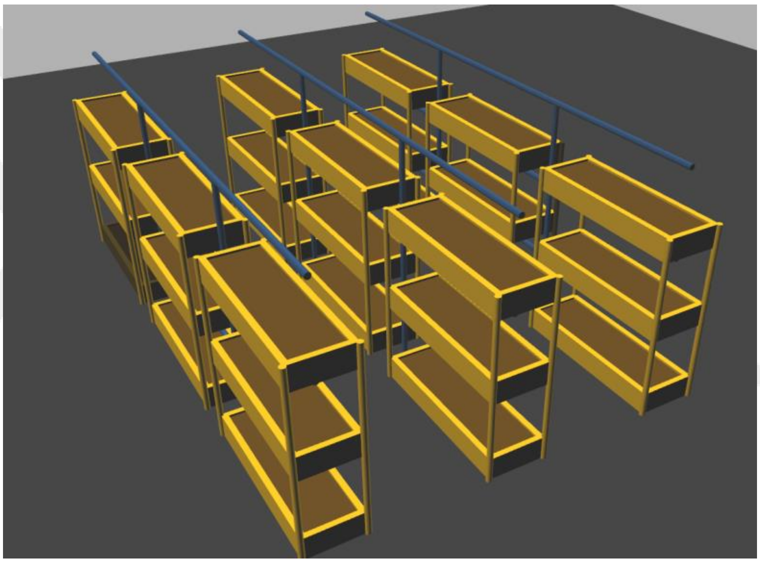

**Figure 2.4 :** Plant racks model

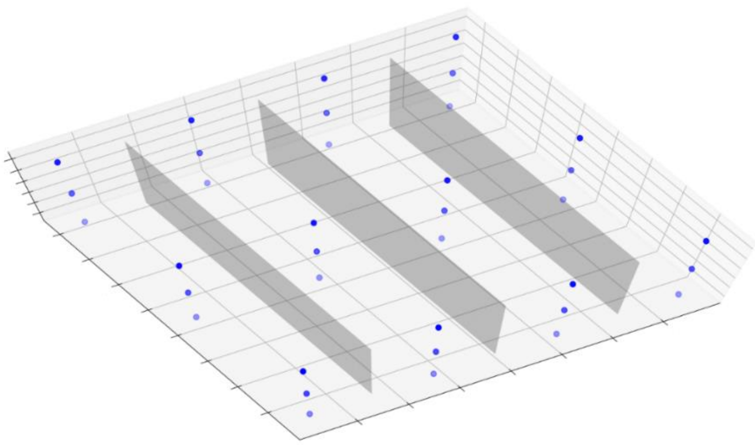

**Figure 2.5 :** Grid points based on the plant racks

ROS integration with Gazebo facilitates seamless communication between different modules in the simulation. ROS handles data flow from the simulated sensors to processing nodes, enabling real-time adjustments. For example, the UAV's path and trajectory planning modules can dynamically adapt to positional data from the simulation, ensuring precise navigation through the greenhouse.

# 3. RESULTS AND DISCUSSION

## 3.1 Path Planning

In this study, various solutions to the traveling salesman problem were evaluated as applied to UAV path planning in complex indoor environments. Different TSP solution strategies, including first solution strategies and local search metaheuristics, were tested under specific constraints such as time limits to determine their effectiveness and reliability.

Given the probabilistic nature of these algorithms, it is important to note that results may not always be reproducible. This variability highlights the necessity for extensive experimentation to understand how different solution methods behave under varying conditions. Experiments were conducted using a pre-set grid point map, representing the greenhouse layout, and distance matrices, formed from points of interest within that layout.

To adequately capture the complexity of modern vertical farms, where plant numbers can reach into the millions, grid maps with up to 40,000 elements were created [55]. This scale allowed for the simulation of realistic environments and the assessment of different path planning strategies. A time limit of 30 seconds was selected for these experiments, balancing the need for timely solutions with the computational demands of the algorithms.

Preliminary experiments revealed that, within the set time limit, the impact of the chosen local search metaheuristic typically overshadowed the influence of the initial solution strategy. To isolate the effects of local search metaheuristics more effectively, initial solution strategies were first compared based on their solution-finding speed. As it can be seen from Table 3.1, Path Cheapest Arc emerged as a consistently fast option across various point counts and was subsequently used as the baseline for further comparisons.

**Table 1.4:** First solution strategy comparison with different point counts

| Point Count | Path Cheapest Arc | Global Cheapest Arc | Local Cheapest Insertion | Savings |
|---|---|---|---|---|
| 50 | 5 | 108 | 17 | 33 |
| 100 | 23 | 536 | 65 | 126 |
| 200 | 85 | 2749 | 263 | 516 |
| 350 | 261 | 9904 | 792 | 1680 |
| 500 | 535 | 22688 | 1585 | 3173 |
| 700 | 1037 | 30000+ | 4219 | 8754 |
| 900 | 2763 | 30000+ | 7277 | 14537 |
| 1250 | 5183 | 30000+ | 14142 | 27726 |

The minimum distance vs time graphs of Patch Cheapest Arc with different local search metaheuristics is given in Figure 3.1. At lower point counts, differences in calculated route distances were negligible, and no stable differences between metaheuristics were observed. However, as the number of points increased, Simulated Annealing, Tabu Search, and Generic Tabu Search demonstrated more stable performance. Additionally, as point numbers increased, the 30-second time limit proved insufficient for solutions to reach a plateau, indicating that more time is required for complex cases.

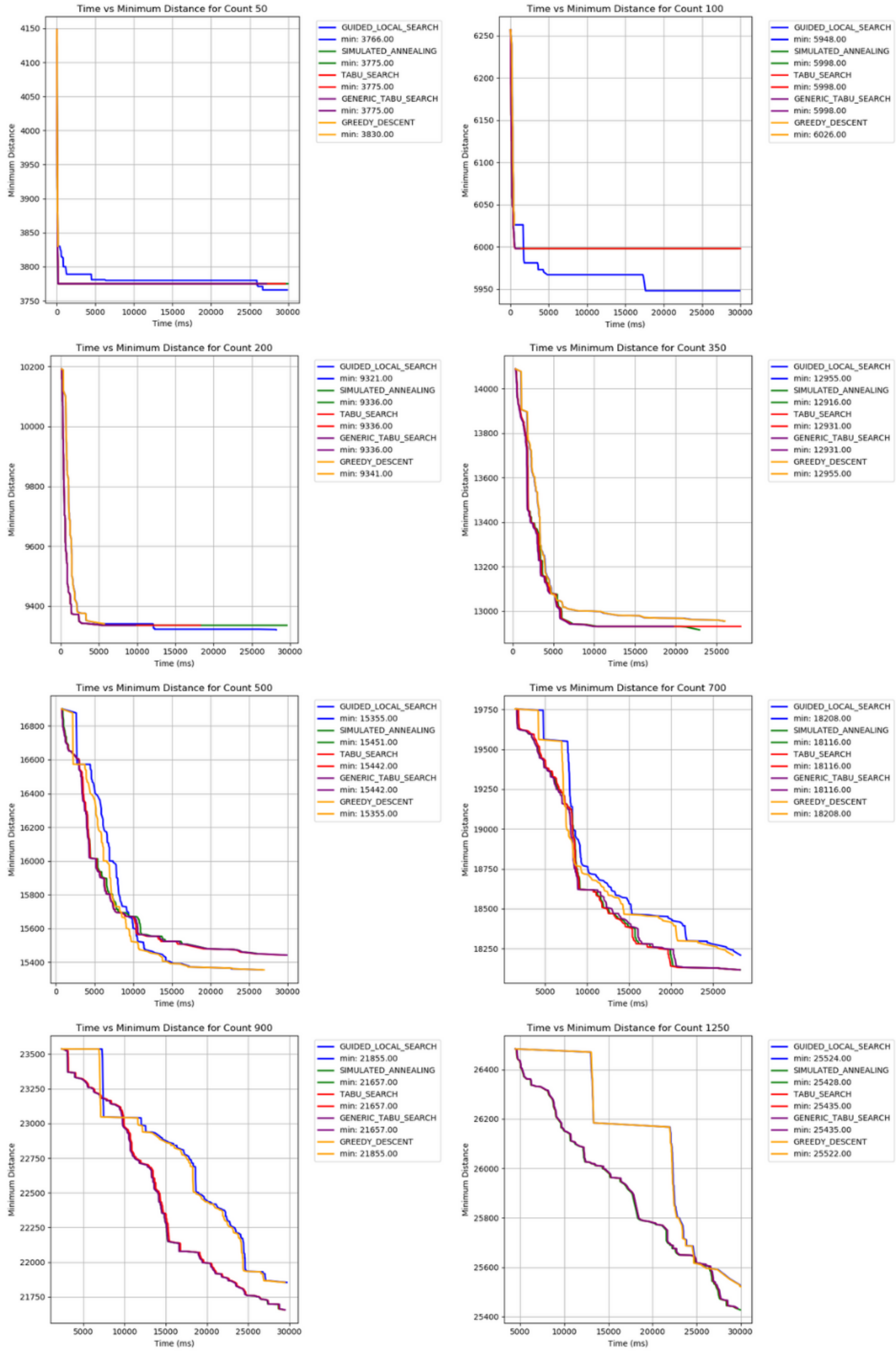


**Figure 3.1 :** Patch chepast arc performance with different point counts and local search metaheuristics

For practical implementation, it was determined that creating the grid map once for each greenhouse is sufficient. After the grid map is created, a dynamic time limit function is employed. This function evaluates different point numbers by running repeated randomized tests for each count. It then selects a time limit for each point count based on the time at which the solution distances start to plateau during these randomized tests. This dynamic adjustment is crucial for maintaining efficiency and effectiveness across varying greenhouse sizes and configurations.

## 3.2 Image Processing

### 3.2.1 Yield estimation

For the yield estimation model, the dataset was divided into training, testing, and validation sets with a split of 80%, 10%, and 10%, respectively. The training process was initially set for 100 epochs; however, it concluded at epoch 67 due to the early stopping mechanism, triggered by the patience parameter set to 10. This mechanism effectively prevented overfitting by halting the training when no improvement in validation metrics was observed for ten consecutive epochs.

The initial learning rate was set to 0.01, to strike a balance between rapid convergence and stability, allowing the model to learn effectively without the risk of overshooting the optimal solution. Additionally, weight decay was set at 0.0005, which penalized large weights and helped prevent overfitting, ensuring the model generalized well on new data.

The model's performance was evaluated using several metrics, including the Recall-Confidence Curve, F1-Confidence Curve, Precision-Confidence Curve, and Precision-Recall Curve. High values in the precision-recall curve indicated that the model effectively identified true positives while minimizing false positives. Balanced F1 scores suggested a good equilibrium between precision and recall, signifying that the model was neither too conservative nor too lenient in its predictions. Consistent performance across different confidence levels in recall, precision, and F1 scores demonstrated the model’s robustness and reliability. These curves are demonstrated on Figure 3.2.

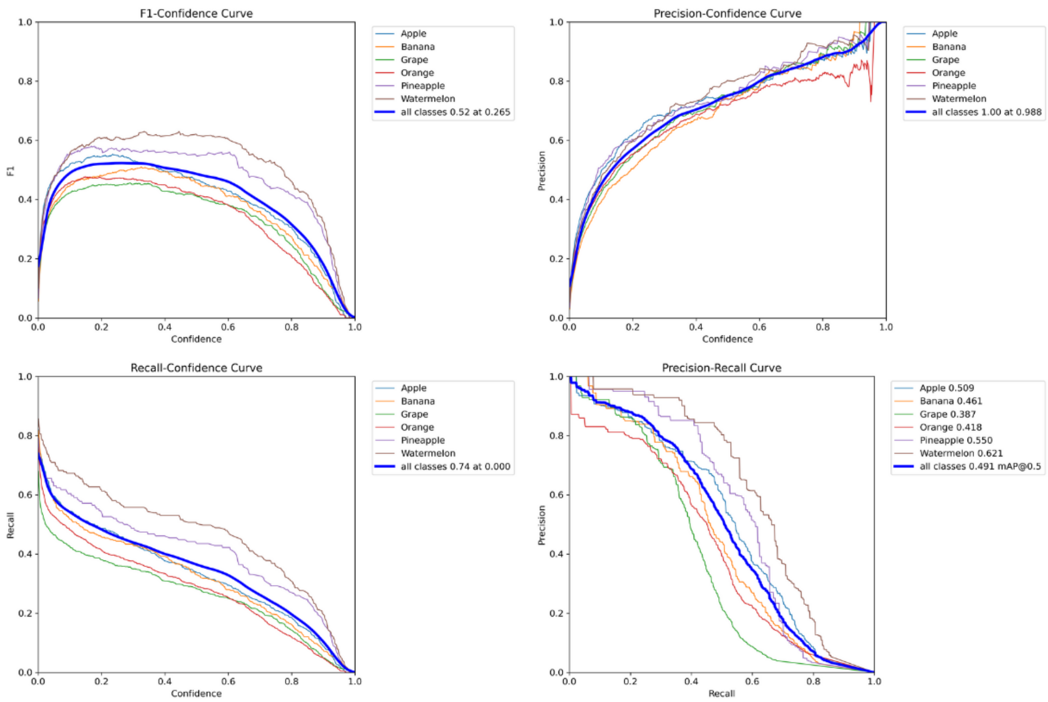


**Figure 3.2 :** Fruit detection model training curves

The training results, which are shown on Figure 3.3, revealed a consistent decrease in both training and validation losses for box regression, classification, and Distribution Focal Loss (DFL), indicating effective learning and convergence. The significant decrease in validation losses underscored the model's ability to generalize well to unseen data. The precision and recall metrics improved steadily, reaching high values, which indicated a balanced performance between true positive detection and the minimization of false positives. The mAP metrics, calculated at IoU thresholds of 0.5 and 0.5-0.95, showed progressive improvement, suggesting the model effectively captured relevant features across various object classes. The smooth training curves, without significant fluctuations, confirmed stable training, free from overfitting or underfitting, highlighting the robustness and effectiveness of the model training process.

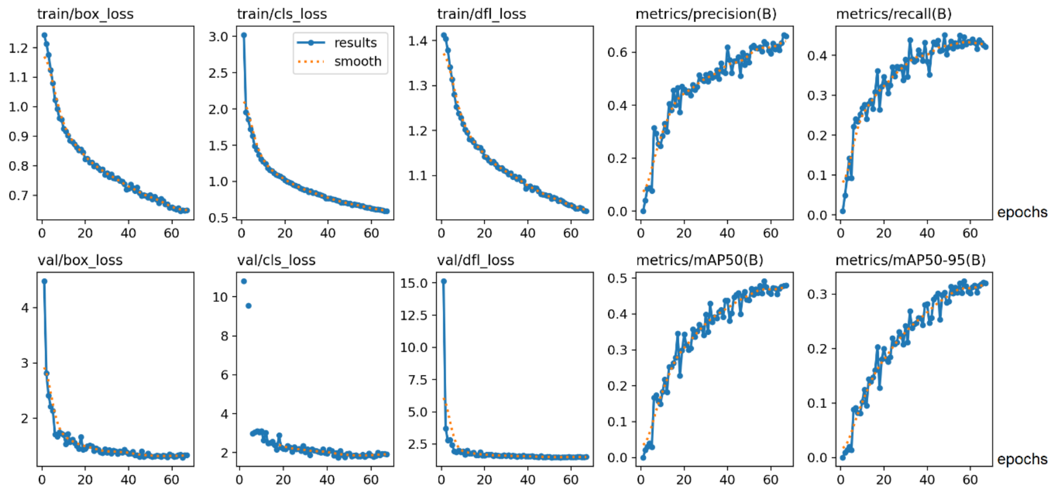


**Figure 3.3 :** Fruit detection model training results

To make sure the detection model is applicable in real-world situations, tests have been carried out on real-world imagery as shown in the Figure 3.4. An analysis of each bounding box's average color is one of the filtration techniques used. Technically, the model calculates the mean color value of the detected region and compares it to a pre-set color range for the target grape type. By using this technique, the system is able to filter out inaccurate detections, such as when green grapes are mistaken with purple ones or when non-grape objects are recognized as grapes. It also makes it possible to identify unripe purple grapes. This color-based filtration improves the overall accuracy and efficiency of the detection process by ensuring that only the desired grape types are identified while lowering the number of false positives.


**Figure 3.4 :** Fruit detection model color filter process

The Figure 3.5 demonstrates the fruit detection model's performance in real-world conditions, highlighting its robustness and practical application. Even with a few very small non-detections, the system performs effectively in terms of correctly identifying grapes overall. This real-world example demonstrates how well the model works in both indoor agricultural setups and well-lit outdoor environments. The successful detection of grapes in these conditions confirms the model's reliability, making it a valuable tool for indoor farmers to estimate yields, and optimize agricultural practices.


**Figure 3.5 :** Fruit detection model real-world results (grape)

The fruit detection model's performance in challenging real-world scenarios is depicted in the Figure 3.6. The original shots were too large to be processed accurately due to the size of the tree, which resulted in cropping that reduced the quality of the image. Furthermore, there is not enough lighting because the leaves are shaded, which is not a problem in a controlled indoor farming environment. In spite of these less-than-ideal conditions, the model exhibits strong performance. Considering the difficult circumstances, the results are impressive overall, with just one false positive and a small number of non-detections. This demonstrates the potential and dependability of the model and implies that the detection accuracy would be even higher in indoor agricultural settings, meaning accurate and efficient crop monitoring.


**Figure 3.6 :** Fruit detection model real-world results (apple)

### 3.2.2 Disease detection

For the disease classification model, an 80/20 split was utilized for the training and testing datasets. With this division the model has sufficient data to learn from while retaining a considerable amount for testing to verify its performance. The same parameters used in the yield estimation model were applied here, including an initial learning rate of 0.01 and a weight decay of 0.0005 to maintain a balance between rapid convergence and model stability. Training concluded at 62 epochs, as the early stopping mechanism halted the process once the validation metrics ceased to improve, preventing overfitting.

The confusion matrix in the Figure 3.7 indicates that the classification model is performing well, as the majority of the predictions fall along the diagonal, representing correct classifications. The minimal off-diagonal values indicate few misclassifications, further supporting the model's effectiveness. The consistent and high diagonal values across different classes show that the model generalizes well and is reliable in distinguishing between categories of plant diseases and health states.

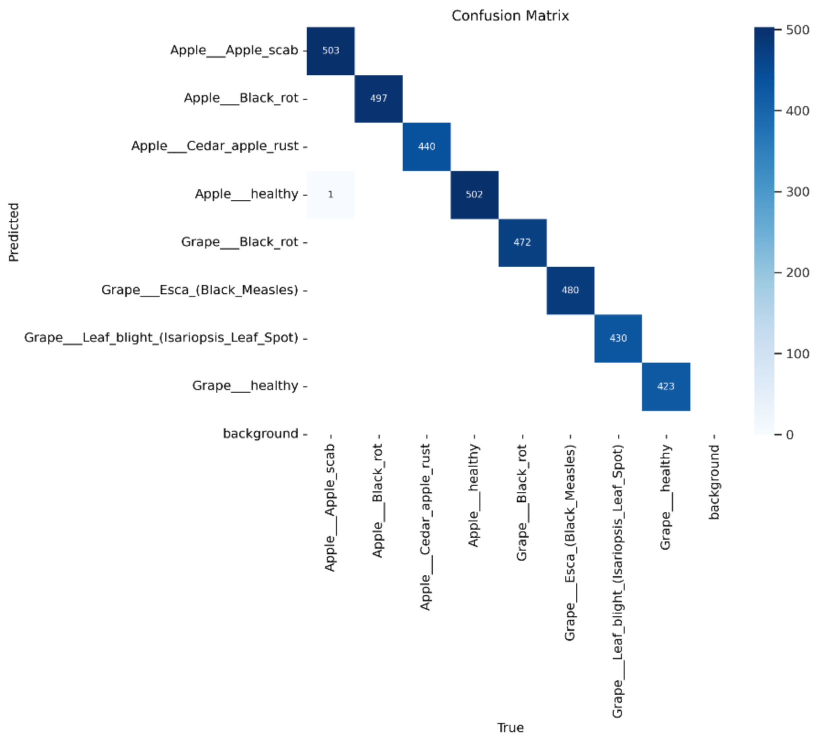


**Figure 3.7 :** Disease classification model confusion matrix

The training and validation loss graphs are shown in Figure 3.8. These graphs reveal a consistent decline, indicating that the model is effectively learning and not overfitting. The top-1 accuracy metric quickly rises and stabilizes near 1.0, demonstrating excellent performance in predicting the correct class. The top-5 accuracy is consistently at 1.0, which, given the model has only 8 classes, means that the correct class is always among the top 5 predictions. This is expected behavior for a small number of classes, as it becomes easier for the true class to appear within the top 5 predictions. Overall, the metrics indicate that the model is well-trained and highly accurate.

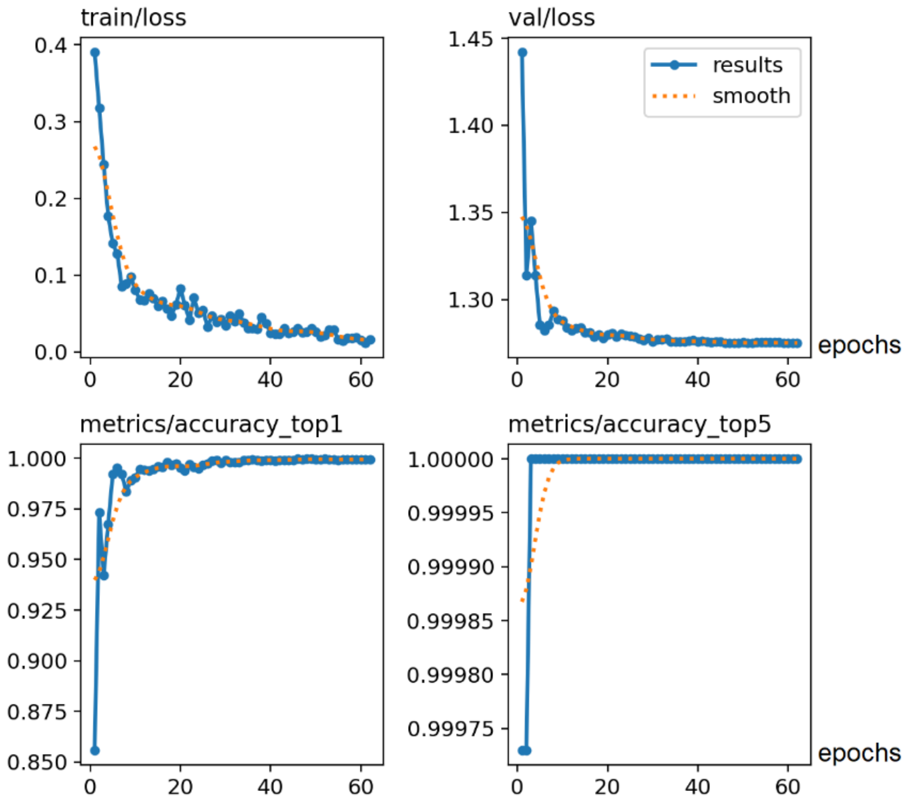


**Figure 3.8 :** Disease classification model training results

The disease classification model's ability to identify and categorize leaves as either healthy or diseased was assessed using real-world images, shown in Figure 3.9. The identified leaves are shown in boxes; diseased leaves are marked in red, and healthy leaves are highlighted in green. Visual inspections show that the classifications are mostly plausible suggesting that the model does an effective job of determining the leaves' state of health. To validate the model's performance and provide a more accurate assessment, it would be necessary to get an expert's opinion on each tested leaf. Furthermore, the leaf detection model performed successfully, precisely recognizing leaves within the pictures.

**Figure 3.9 :** Disease classification model real-world results (grape)

The disease classification model's performance was also assessed under poor conditions, akin to those previously discussed. Figure 3.10 presents these findings. Some leaves were not detected, and the classification model might have incorrectly classified some leaves due to factors like reduced image quality from cropping and inadequate lighting producing shadows. Overall, the results are still promising despite these difficulties. The model's performance in less-than-ideal conditions indicates that it would perform much better in a controlled indoor farm environment. However, expert validation—possibly from a botanist—is advised to confirm the health status of the detected leaves and offer further insights for model improvement in order to ensure the reliability of the classifications.

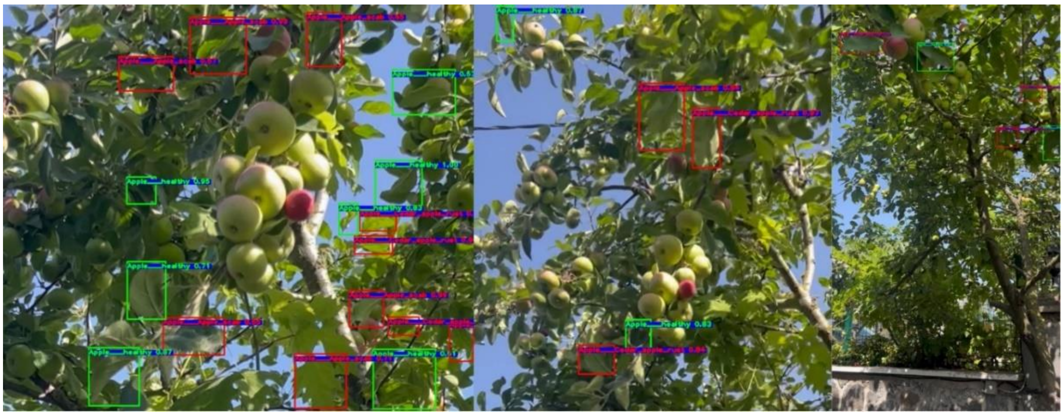

**Figure 3.10 :** Disease classification model real-world results (apple)

### 3.3 System Integration and Simulation

During the simulation in the Gazebo environment, the UAV executed the generated flight paths accurately, demonstrating smooth and stable movements throughout the indoor agricultural setup. A sample image taken from the simulation, capturing the moment when the UAV is processing an image, is presented in Figure 3.11.

**Figure 3.11 :** UAV processing the image of a branch during simulation run

The images captured by the UAV's camera were processed in real-time by the Image Processing module. The annotated images provided clear and valuable information regarding plant health and yield estimates. The system's ability to detect diseased plants and estimate yields was validated through these simulations, showcasing high accuracy and reliability, despite a few minor misfires. These inaccuracies were minimal and did not significantly affect the overall performance. The annotated images confirmed the correct identification of disease symptoms and accurate yield assessments, aligning closely with the expected outcomes of the image processing algorithms. 4 samples of these annotated images are shared in Figure 3.12.

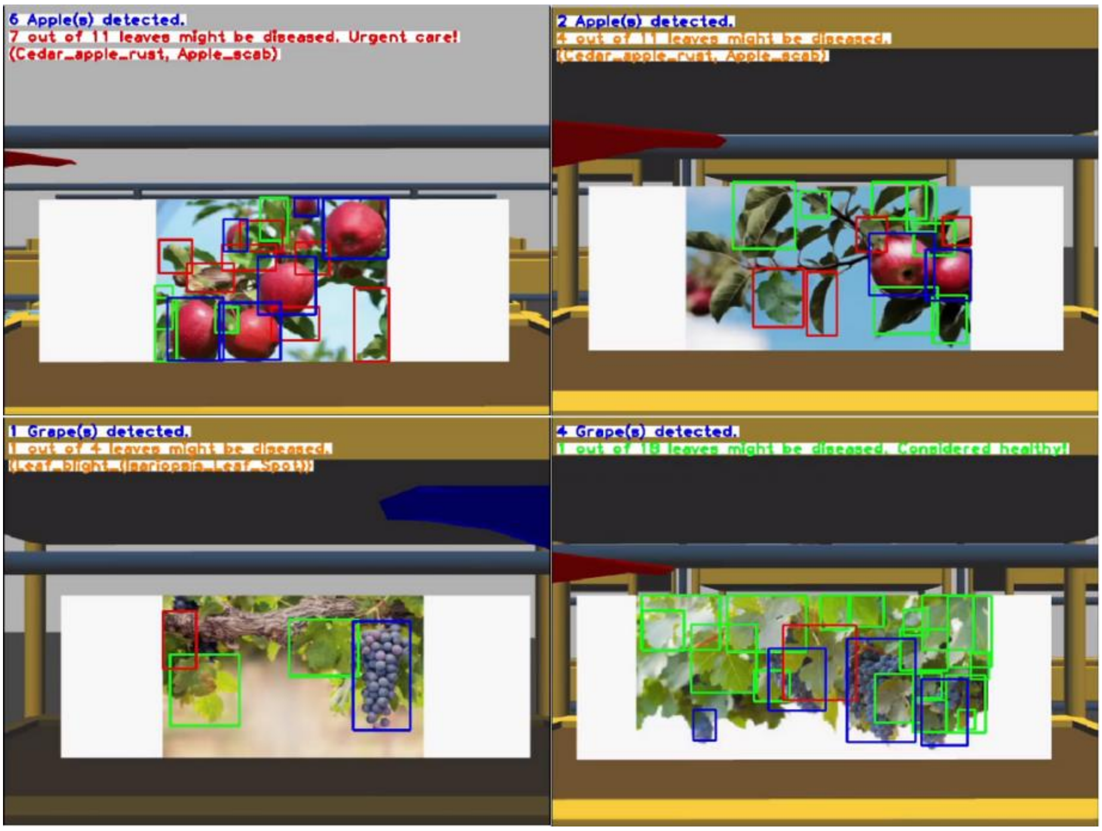


**Figure 3.12 :** Annotated images from simulation

# 4. CONCLUSIONS AND FUTURE WORK

This study has demonstrated the effective integration of UAVs with advanced path planning and image processing techniques to advance indoor agriculture. By addressing the challenges of disease detection and yield estimation, the research contributes significantly to the field of CEA. The proposed system, which combines path planning algorithms tailored for confined spaces and deep learning models for image analysis, showed promising results in simulated environments with real-world imagery. This modular and adaptable approach can be seamlessly integrated into existing indoor farming operations, potentially transforming the efficiency and sustainability of indoor agriculture. The advancements made are crucial for optimizing resource allocation, improving crop management, and ensuring a stable food supply. Moreover, the use of the ROS and Gazebo simulation platform facilitated the development and testing of the system in a controlled and scalable manner, reducing the risks and costs associated with physical trials.

While this research has took significant steps, several areas might need further investigation. Firstly, real-world deployment and scalability need to be thoroughly evaluated. Conducting field trials in diverse indoor farming setups will help refine the system's robustness and adaptability. Additionally, exploring the integration of more advanced sensors, such as LIDARs, and improving the UAV's flight dynamics can enhance data accuracy and operational efficiency. Future research should also focus on collaborating with experts to validate the disease detection model and extend the application to other crops and diseases.

## REFERENCES


[1] **Hunter, M. C., Smith, R. G., Schipanski, M. E., Atwood, L. W., & Mortensen, D. A.** (2017). Agriculture in 2050: Recalibrating Targets for Sustainable Intensification. BioScience/Bioscience, 67(4), 386–391. https://doi.org/10.1093/biosci/bix010

[2] **Foley, J. A., Ramankutty, N., Brauman, K. A., Cassidy, E. S., Gerber, J. S., Johnston, M., Mueller, N. D., O'Connell, C., Ray, D. K., West, P. C., Balzer, C., Bennett, E. M., Carpenter, S. R., Hill, J., Monfreda, C., Polasky, S., Rockström, J., Sheehan, J., Siebert, S., . . . Zaks, D. P. M.** (2011). Solutions for a cultivated planet. Nature, 478(7369), 337–342. https://doi.org/10.1038/nature10452

[3] **Benke, K., & Tomkins, B.** (2017). Future food-production systems: vertical farming and controlled-environment agriculture. Sustainability: Science, Practice, & Policy, 13(1), 13–26. https://doi.org/10.1080/15487733.2017.1394054

[4] **Cho, R.** (2011, October 13). Vertical Farms: From Vision to reality. State of the Planet. https://news.climate.columbia.edu/2011/10/13/vertical-farms-from-vision-to-reality/ Retrieved 2024-05-19.

[5**] Naresh, Ram & J, Sagar & Beese, Shreedhar & singh, Monika.** (2024). Role of Hydroponics in Improving Water-Use Efficiency and Food Security. International Journal of Environment and Climate Change. 14. 608-633. 10.9734/IJECC/2024/v14i23976.

[6**] Roberts, J. M., Bruce, T. J. A., Monaghan, J. M., Pope, T. W., Leather, S. R., & Beacham, A. M.** (2020). Vertical farming systems bring new considerations for pest and disease management. Annals of Applied Biology/Annals of Applied Biology, 176(3), 226–232. https://doi.org/10.1111/aab.12587

[7] **Zhang, C., Atkinson, P. M., George, C., Wen, Z., Diazgranados, M., & Gerard, F.** (2020). Identifying and mapping individual plants in a highly diverse high-elevation ecosystem using UAV imagery and deep learning. ISPRS Journal of Photogrammetry and Remote Sensing, 169, 280–291. https://doi.org/10.1016/j.isprsjprs.2020.09.025

[8**] Schiefer, F., Kattenborn, T., Frick, A., Frey, J., Schall, P., Koch, B., & Schmidtlein, S.** (2020). Mapping forest tree species in high resolution UAV-based RGB-imagery by means of convolutional neural networks. ISPRS Journal of Photogrammetry and Remote Sensing, 170, 205–215. https://doi.org/10.1016/j.isprsjprs.2020.10.015

[9] **Faiçal, B. S., Freitas, H., Gomes, P. H., Mano, L. Y., Pessin, G., De Carvalho, A. C., Krishnamachari, B., & Ueyama, J.** (2017). An adaptive approach for UAV-based pesticide spraying in dynamic environments. Computers and Electronics in Agriculture, 138, 210–223. https://doi.org/10.1016/j.compag.2017.04.011

[10] **Meng, Y., Su, J., Song, J., Chen, W., & Lan, Y.** (2020). Experimental evaluation of UAV spraying for peach trees of different shapes: Effects of

operational parameters on droplet distribution. Computers and Electronics in Agriculture, 170, 105282. https://doi.org/10.1016/j.compag.2020.105282

[11] **Fu, Z., Jiang, J., Gao, Y., Krienke, B., Wang, M., Zhong, K., Cao, Q., Tian, Y., Zhu, Y., Cao, W., & Liu, X.** (2020). Wheat Growth Monitoring and Yield Estimation based on Multi-Rotor Unmanned Aerial Vehicle. Remote Sensing, 12(3), 508. https://doi.org/10.3390/rs12030508

[12] **Cao, Y., Li, G. L., Luo, Y. K., Pan, Q., & Zhang, S. Y.** (2020). Monitoring of sugar beet growth indicators using wide-dynamic-range vegetation index (WDRVI) derived from UAV multispectral images. Computers and Electronics in Agriculture, 171, 105331. https://doi.org/10.1016/j.compag.2020.105331

[13] **Romero, M., Luo, Y., Su, B., & Fuentes, S.** (2018). Vineyard water status estimation using multispectral imagery from an UAV platform and machine learning algorithms for irrigation scheduling management. Computers and Electronics in Agriculture, 147, 109–117. https://doi.org/10.1016/j.compag.2018.02.013

[14] **Dos Santos Ferreira, A.; Freitas, D.M.; da Silva, G.G.; Pistori, H.; Folhes, M.T**. (2017) Weed detection in soybean crops using ConvNets. Comput. Electron. Agric. 143, 314–324.

[15] **Wan, L., Cen, H., Zhu, J., Zhang, J., Zhu, Y., Sun, D., Du, X., Zhai, L., Weng, H., Li, Y., Li, X., Bao, Y., Shou, J., & He, Y.** (2020). Grain yield prediction of rice using multi-temporal UAV-based RGB and multispectral images and model transfer – a case study of small farmlands in the South of China. Agricultural and Forest Meteorology, 291, 108096. https://doi.org/10.1016/j.agrformet.2020.108096

[16] **Tokekar, P., Hook, J. V., Mulla, D., & Isler, V.** (2016). Sensor planning for a symbiotic UAV and UGV system for precision agriculture. IEEE Transactions on Robotics, 32(6), 1498–1511. https://doi.org/10.1109/tro.2016.2603528

[17] **Johansen, K., Duan, Q., Tu, Y., Searle, C., Wu, D., Phinn, S., Robson, A., & McCabe, M. F.** (2020). Mapping the condition of macadamia tree crops using multi-spectral UAV and WorldView-3 imagery. ISPRS Journal of Photogrammetry and Remote Sensing, 165, 28–40. https://doi.org/10.1016/j.isprsjprs.2020.04.017

[18] **Mohanty, S. P., Hughes, D. P., & Salathé, M.** (2016). Using deep learning for Image-Based plant disease detection. Frontiers in Plant Science, 7. https://doi.org/10.3389/fpls.2016.01419

[19] **Garcia-Ruiz, F., Sankaran, S., Maja, J. M., Lee, W. S., Rasmussen, J., & Ehsani, R.** (2013). Comparison of two aerial imaging platforms for identification of Huanglongbing-infected citrus trees. Computers and Electronics in Agriculture, 91, 106–115. https://doi.org/10.1016/j.compag.2012.12.002

[20] **Sarkar, S. K., Das, J., Ehsani, R., & Kumar, V.** (2016). Towards autonomous phytopathology: Outcomes and challenges of citrus greening disease detection through close-range remote sensing. In 2016 IEEE International Conference on Robotics and Automation (ICRA). https://doi.org/10.1109/icra.2016.7487719

[21] **Kerkech, M., Hafiane, A., Canals, R., Ros, F.** (2020). Vine Disease Detection by Deep Learning Method Combined with 3D Depth Information. In:

El Moataz, A., Mammass, D., Mansouri, A., Nouboud, F. (eds) Image and Signal Processing. ICISP 2020. Lecture Notes in Computer Science(), vol 12119. Springer, Cham. https://doi.org/10.1007/978-3-030-51935-3_9

[22] **Kerkech, M., Hafiane, A., & Canals, R.** (2018). Deep learning approach with colorimetric spaces and vegetation indices for vine diseases detection in UAV images. Computers and Electronics in Agriculture, 155, 237–243. https://doi.org/10.1016/j.compag.2018.10.006

[23] **Di Gennaro, S., Battiston, E., Di Marco, S., Facini, O., Matese, A., Nocentini, M., Palliotti, A., & Mugnai, L.** (2016). Unmanned Aerial Vehicle (UAV)-based remote sensing to monitor grapevine leaf stripe disease within a vineyard affected by esca complex. Phytopathologia Mediterranea, 55(2), 262–275. https://doi.org/10.14601/Phytopathol_Mediterr-18312

[24] **Zhang, K., Wu, Q., Liu, A., & Meng, X.** (2018). Can deep learning identify tomato leaf disease? Advances in Multimedia, 2018, 1–10. https://doi.org/10.1155/2018/6710865

[25] **Abdulridha, J., Ampatzidis, Y., Kakarla, S. C., & Roberts, P.** (2019). Detection of target spot and bacterial spot diseases in tomato using UAV-based and benchtop-based hyperspectral imaging techniques. Precision Agriculture, 21(5), 955–978. https://doi.org/10.1007/s11119-019-09703-4

[26] **Url-1** Tevel Tech. https://www.tevel-tech.com/ Retrieved 2024-07-22

[27] **Roldán, J. J., Joossen, G., Sanz, D., Del Cerro, J., & Barrientos, A.** (2015). Mini-UAV based sensory system for measuring environmental variables in greenhouses. Sensors, 15(2), 3334–3350. https://doi.org/10.3390/s150203334

[28] **Simon, J., Petkovic, I., Petkovic, D., & Petkovics, Á.** (2018). Navigation and applicability of hexa rotor drones in greenhouse environment. Tehnički Vjesnik/TehničKi Vjesnik, 25(Supplement 2). https://doi.org/10.17559/tv-20161109211133

[29] **Land, A. H., & Doig, A. G.** (1960). An automatic method of solving discrete programming problems. Econometrica, 28(3), 497. https://doi.org/10.2307/1910129

[30] **Held, M., & Karp, R. M.** (1962). A Dynamic Programming Approach to Sequencing Problems. Journal of the Society for Industrial and Applied Mathematics, 10(1), 196–210. https://doi.org/10.1137/0110015

[31] **Schrijver, A.** (2005). On the History of Combinatorial Optimization (Till 1960). In Handbooks in operations research and management science (pp. 1–68). https://doi.org/10.1016/s0927-0507(05)12001-5

[32] **Goodrich, Michael T.; Tamassia, Roberto** (2015), "18.1.2 The Christofides Approximation Algorithm", Algorithm Design and Applications, Wiley, pp. 513–514.

[33] **Mitchell, M.** (1998). An introduction to genetic algorithms. https://doi.org/10.7551/mitpress/3927.001.0001

[34] **Colorni, A., Dorigo, M., Maniezzo, V., Varela, F.J., & Bourgine, P.E.** (1992). Distributed Optimization by Ant Colonies.

[35**] Url-2** "What is Simulated Annealing?". www.cs.cmu.edu. Retrieved 2024-05-19.

[36] **Laurent Perron and Vincent Furnon** OR-Tools v9.9. https://developers.google.com/optimization/ Retrieved 2024-05-19.

[37] **Clarke, G., & Wright, J. W.** (1964). Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. Operations Research, 12(4), 568–581. https://doi.org/10.1287/opre.12.4.568
[38] **Voudouris, C., Tsang, E.P.K.** (2003). Guided Local Search. In: Glover, F., Kochenberger, G.A. (eds) Handbook of Metaheuristics. International Series in Operations Research & Management Science, vol 57. Springer, Boston, MA. https://doi.org/10.1007/0-306-48056-5_7
[39] **Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P.** (1983). Optimization by simulated annealing. Science, 220(4598), 671–680. https://doi.org/10.1126/science.220.4598.671
[40**] Glover, F.** (1986). Future paths for integer programming and links to artificial intelligence. Computers & Operations Research, 13(5), 533–549. https://doi.org/10.1016/0305-0548(86)90048-1
[41] **Girshick, R.B.** (2015). Fast R-CNN.
[42] **Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S.E., Fu, C., & Berg, A.C.** (2015). SSD: Single Shot MultiBox Detector. European Conference on Computer Vision.
[43] **Zhou, X., Wang, D., & Krähenbühl, P.** (2019). Objects as Points. ArXiv, abs/1904.07850.
[44] **Jocher, G., Chaurasia, A., & Qiu, J.** (2023). Ultralytics YOLO (Version 8.0.0) [Computer software]. https://github.com/ultralytics/ultralytics
[45] **Lakshay Tyagi.** (2023). Fruit Detection Dataset [Data set]. Kaggle. https://doi.org/10.34740/KAGGLE/DSV/4922010
[46] **Krizhevsky, A., Sutskever, I., & Hinton, G.E.** (2012). ImageNet classification with deep convolutional neural networks. Communications of the ACM, 60, 84 - 90.
[47] **Simonyan, K., & Zisserman, A.** (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. CoRR, abs/1409.1556.
[48] **Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S.E., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A.** (2014). Going deeper with convolutions. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1-9.
[49] **He, K., Zhang, X., Ren, S., & Sun, J.** (2015). Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770-778.
[50] **Url-3** FODUUCOM/plant-leaf-detection-and-classification. foduucom/plant-leaf-detection-and-classification · Hugging Face. (n.d.). https://huggingface.co/foduucom/plant-leaf-detection-and-classification Retrieved 2024-05-19
[51] **Url-4** New plant diseases Dataset. (2018, November 18). Kaggle. https://www.kaggle.com/datasets/vipoooool/new-plant-diseases-dataset/data Retrieved 2024-05-19
[52] **Markovic, L., Petric, F., Ivanovic, A., Goricanec, J., Car, M., Orsag, M., & Bogdan, S.** (2023). Towards A Standardized Aerial Platform: ICUAS'22 Firefighting Competition. Journal of Intelligent & Robotic Systems, 108(3), 52. doi:10.1007/s10846-023-01909-z
[53] **Quigley, M.** (2009). ROS: an open-source Robot Operating System. IEEE International Conference on Robotics and Automation.
[54] **Koenig, N., & Howard, A.** (2004). Design and use paradigms for Gazebo, an open-source multi-robot simulator. In 2004 IEEE/RSJ International

Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566) (pp. 2149-2154 vol.3).

[55] **Marks, P.** (2014). Vertical farms sprouting all over the world. New Scientist. https://www.newscientist.com/article/mg22129524-100-vertical-farms-sprouting-all-over-the-world/ Retrieved 2024-05-19

**CURRICULUM VITAE**

**Name Surname** **:** Onat Erdoğmuş

**EDUCATION** **:**

- **B.Sc.** **:** 2020, Istanbul Technical University, Faculty of Mechanical Engineering, Mechanical Engineering Program

**PROFESSIONAL EXPERIENCE AND REWARDS:**

- 2023 – Present, Embedded Software Engineer, Ford Otosan
- 2021 – 2023, System Design Engineer, Arçelik

**PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS:**

- **Onat Erdoğmuş,** Erdinç Altuğ (2024). "Integrating Path Planning and Image Processing with UAVs for Disease Detection and Yield Estimation in Indoor Agriculture", 3rd International Graduate Research Symposium (IGRS'24), May 8-10, 2024 İstanbul – TÜRKİYE.