UNIVERSITÀ DEGLI STUDI DI NAPOLI "FEDERICO II"

PH.D. THESIS

XXXIV CYCLE

SUSTAINABLE AGRICULTURAL AND FORESTRY SYSTEMS AND FOOD SECURITY

VISmaF: Immersive Virtual Visualization in Smart Farming and Digital Advances in Agriculture

---

Author:
Eng. Mariano CRIMALDI

Supervisor:
Prof. Francesco GIANNINO

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

in the

Department of Agricultural Sciences

2018 - 2022

Declaration of Authorship

I, Eng. Mariano CRIMALDI, declare that this thesis titled, "VISmaF: Immersive Virtual Visualization in Smart Farming and Digital Advances in Agriculture" and the work presented in it are my own. I confirm that:

- This work has been supported and funded by the P.O.R. Campania FSE 2014/2020 "*Dottorati di ricerca con caratterizzazione industriale*" grant.
- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: 

Date: 07 March 2022

"Daaaa daaaa daaaa maaaaaa maaaa maaaa mmaaa!"

My son's first argument about life, the universe and everything else…

Contents

List of Figures

List of Tables

List of code listings

List of Abbreviations

DT	**D**igital **T**win
ODE	**O**rdinary **D**ifferential **E**quation
OOP	**O**bject **O**riented **P**rogramming
FSPM	**F**unctional **S**tructural **P**lant **M**odel
FPM	**F**unctional **P**lant **M**odel
SPM	**S**tructural **P**lant **M**odel
RGG	**R**elational **G**rowth **G**rammars
GU	**G**rowth **U**nits
FU	**F**ruiting **U**nits
PAR	**P**hotosynthetic **A**ctive **R**adiation
GI	**G**lobal **I**llumination
TRV	**T**ree **R**ow **V**olume
UAV	**U**nmanned **A**erial **V**ehicle
UTV	**U**nmanned **T**errestrial **V**ehicle
LAI	**L**eaf **A**rea **I**ndex
NDVI	**N**ormalized **D**ifference **V**egetation **I**ndex
IWM	**I**ntegrated **W**eed **M**anagement
SSWM	**S**ite **S**pecific **W**eed **M**anagement
AWR	**A**utonomous **W**eeding **R**obot

Chapter 1

Introduction

Digital technologies have taken hold over the past few decades in many scientific and industrial sectors: from the aviation industry to medicine, from engineering to agriculture. Thanks to the rapid development of hardware and software solutions, it has been possible to implement techniques and technologies that were unthinkable until recently. A striking example is the adoption of the *Digital Twin* (DT), which is a digital counterpart, perfectly reproduced, of the object or model that is to be studied [1]. The adoption of a DT allows the study, through numerical models, of complex systems without having to materially build, implement or in some cases destroy them, simplifying the search for new solutions also from an economic point of view [2]. The digitization of processes and their scientific study has also been introduced in agriculture where today it is possible to adopt advanced systems for remote sensing, management of fields or farms, decision support [3]. In the specific case of crop modeling, great advances have been made in the last decade thanks to the development of the so-called FSPM, or biological/mathematical models that take into account the functional processes of a plant, in connection and in relation to its geometric structure [4, 5]. These models allow the creation of a particular DT called *plant in silico* that is a digital plant with which it is possible to understand, study and visualize all the possible changes made at physiological, genetic, phenological and other levels [6, 7]. *In silico* approaches can also be used at a larger scale, at the stand level or on a territorial scale, to study broader phenomena. Also at the stand or spatial level, the introduction of remote sensing in agriculture has allowed further digitization of the field. The introduction in agriculture in recent years of increasingly accessible and easy-to-use systems, thanks to sensors mounted on Unmanned Aerial Vehicles, has allowed frequent, robust and accurate data collection. These data are used in a variety of ways: from creating vegetation indices to obtain prescription maps, to using them in Deep Learning algorithms for automatic recognition of diseases, phenological status, land uses and more. The present study aimed to develop advanced digitization techniques for some applications in the field of agriculture with the goal of presenting and deploy some products used in concrete applications by the industry of the sector. In the first chapter (2), the state of the art of FSPM models has been reviewed, in particular those with a 3D tree structure output. This review showed a lack of system-based models so a proof of concept of a

biological/mathematical system-dynamics model coupled with a real-time 3D rendering engine has been proposed. This solution has been studied because there is a rapidly expanding industry sector that is, as a particular DT, process virtualization. In agriculture, process virtualization, unlike the process industry, is mostly absent. The study therefore proposes itself as a possible model of virtualization of biologically verisimilar trees (the aforementioned *in silico* plants) in order to be able to use them in industrial agronomic applications, such as training of agricultural operators, without having to physically grow, modify or perhaps destroy a real tree. This proof of concept has been then developed in detail in chapter 3 where it has been explained more in detail, developed with its characteristics, its code, its implementation and its possible agronomic applications. A Deep Learning system for weed recognition has been developed in chapter 4. This study has been carried out in order to have an automated aerial (UAV) or terrestrial (UTV) system that allows selective recognition of weeds to be removed and not to be removed in order to provide farmers or the industry with a Site-Specific Weed Management tool that allows them to save on herbicide use. Finally, in chapter 5, a study has been developed on the interaction between water flows and plants in vegetated channels in addition to other digital agriculture applications for eco-hydraulics applications. The study has been possible thanks to the adoption of some digital techniques such as 3D scanning of plants to create a DT used in numerical finite element simulations. In addition, Remote Sensing approaches have been developed using multispectral sensors on drones to calculate the Leaf Area Index (LAI) in hard-to-access vegetated channels and assess the vegetation cover of the channel. This study on vegetated channels has been carried out because the results obtained can be used by the mechanical, aerospace or naval industry to produce vehicles optimized to work in conditions of high turbulence, or more in a smaller sense to manage vegetated channels to optimize hydraulic flows without compromising the plant ecosystem. The projects presented in this paper show how the digitization of agriculture is a broad field, both in terms of applications and theory. The interest shown by industry in these applications raises hopes for rapid future development in what are the different fields of application.

Chapter 2

Biologically aware generation of 3D tree models: review of FSPM models and scientific proposal

2.1 Introduction

Computer simulations of plant growth have a long history. Botanists and computer scientists have tried to develop methods to generate natural objects synthetically for over forty years [8, 9]. The first method for branching structure simulation using a computer called a cellular automaton was proposed by Ulam [10] in 1966: an arrangement of square cells on a grid, directed by a computer program that had the ability to adapt to different conditions. Greene [11] extended the cellular automaton to a three-dimensional space called *voxel space automata* trying to simulate growing processes with reactions and adaptations to the environment. Simpler models, which ignored even fundamental factors such as branch collisions, were developed later. Honda [12] proposed a model in which a variety of tree-like shapes was obtained by changing few numeric parameters. Lindenmayer [13] proposed a string rewriting system for cellular interaction commonly called the L-system, later adopted to generate plants and trees with the contribution of Prusinkiewicz [14].

Oppenheimer [15] used fractals to form trees with more advanced parameters such as branching angle, branch-to-parent size ratio, stem taper rates, helical twist, and branches per stem segment. Oppenheimer introduced random variations to alleviate some of the self-similarities intrinsic in fractal models (same specifications for every recursive level). Other models were developed during the years, following the needs of both 3D artists or professionals and botanists [16].

Following the development of these pioneering methodologies, the field has been revolutionized in the last two decades thanks to the exponential rise of computational capabilities and the integration with mathematical-biological models. The Functional–Structural Plant Models (FSPMs) in the last two decades have been developed by scientists in order to explore and integrate the structure and the underlying processes of a plant [4]. Nowadays, the complexity of digital design has reached elevated levels of quality standards allowing the use of 3D structure as output of

FSPMs in order to characterize plant phenotypes [17] or as feedback to assess and calculate light partitioning [18]. 3D plants can be part of several modeling systems, being main components of many digital representations of landscapes or natural scenarios [19]. As proposed by Pirk et al. [20], a 3D tree can be used in different scientific applications, such as synthetic silviculture [21], generic digital representation of real world [22], flow dynamics [23] and, as mentioned, in botany to determine physiological parameters [6]. In botany, models are used to determine physiological parameters [24, 25, 26]. Geometrical plant modeling allows researchers to validate visually biological processes, such as the interaction of plants with light and environment. In ecology, plant models are of interest to visualize information of deep-lying processes, allowing scientists to see "invisible" things [6]. Examples are represented by how plants develop in reaction to disease or stress and how plants grow after pruning [27, 28]. This is critical to study the competition of plants for light or to estimate field crop canopy photosynthesis [29, 30, 31] or carbon and water flows [32], making an observer aware of the system's mechanism.

In this chapter the author reviews the different modeling approaches of tree generation using FSPM that have 3D output to understand the state of the art of current modeling approaches and how these models can benefit, if any and if functional to the model, from the generated 3D structure. The author decided to focus the review on models that describe the epigeal part of the plant by excluding those that model only the root system. In subsection 2.2 the FSPM methods with 3D output are presented and reviewed, grouped in three categories, depending on model category: FSPM (Functional-Structural Plant Model, subsection 2.2.1), FPM (Functional Plant Model, subsection 2.2.2), SPM (Structural Plant Model, subsection 2.2.3). In subsection 2.2.4 the models described are compared according to their applications in agronomy: which plant organs were modeled and considered, whether plant-plant interaction in community is modeled, whether plant-soil interaction is modeled, what processes were considered (e.g., carbon allocation, photosynthesis, and so on), and the various agronomic and scientific applications.

As stated by Louarn and Song [4], one of the most promising horizon and opportunity of FSPMs is the use of non-invasive imaging and model based methods. The author gives particular emphasis on their use of 3D model, how the 3D models generated are validated and what are the scientific applications. In subsection 2.3 the author provides a brief description of 3D rendering techniques used to generate synthetic trees and how these are used in commercial software. It is also provided a proof of concept in subsection 2.4 for a link between a biological mathematical model based on Ordinary Differential Equations (ODE) and a 3D rendering engine. This link can operate in both ways, i.e., by rendering a tree using parameters from ODE solutions, or by considering geometrical 3D data (e.g. the amount of light from the scene) as input to solve ODEs.

The in-depth development, results of the model and potential agronomic developments and applications are provided in chapter 3

2.2 Plant and Crops Models

The development of computer models of plant functioning and growth has been in scientists' interests in the last 40 years [33]. During the last 20 years, scientists approached a new method to model plants by creating and developing FSPMs [4]. The main idea of this approach is to describe and model the plant as a population of interconnected parts, ranging from the smallest scale (organism prospective and genetic models) to the biggest one (community of plants and ecological models) [34]. Achieving this goal required a number of studies that made concepts such as those of plant architecture widely recognized in botany and the scientific community [35, 36]. In parallel to the development of globally recognized models, the advancement of computing power by personal computers has allowed the development of the first methods and standards. The FSPM models allow a creation of "virtual" plants thanks also to the possibility, given the presence of architecture models, to have as output 3D models of plants [9]. This output, in addition to being visually immediate for the evaluation of the biological model itself [4], allows a more precise calculation of a series of interactions both with the external environment (virtual and not) such as the amount of light intercepted [37, 38], the competition of plants in the community [21] or external mechanical actions (e.g., pruning) [27] and with other parts of the model of smaller scale [39], such as the reaction of the plant to diseases [28]. This study focuses the attention on functional–structural models with 3D plant architecture output. The studies reviewed in this section are grouped in three categories, according to their modeling goal:

- FSPM: models that have both functional and structural modeling;
- FPM: models that have functional modeling;
- SPM: models that have only structural modeling

All the models reviewed have a 3D plant output. The objective of this review chapter is to assess how the 3D output of models is generated, how it is useful for the development of the model itself, how the 3D structure could be improved, and how it could be a new opportunity for the further development of FSPM models.

All the models reviewed are shown in table 2.1 with their names, the programming language used, and the modeling environment. The works reviewed have been selected through a search on major public databases (*Scopus*, *Web of Science*, *ScienceDirect*, *SpringerLink*, *ACM Digital Library*), among those that have 3D output and that have a particular feedback in the scientific community in terms of citations and applications.

Another table (Table 2.2) is presented in subection 2.2.4 to summarize the agronomic applications, modeling objectives, modeled organs, and whether plant–plant and plant–soil interactions are present in order to have an overview of the described models and proceed to critically compare them.

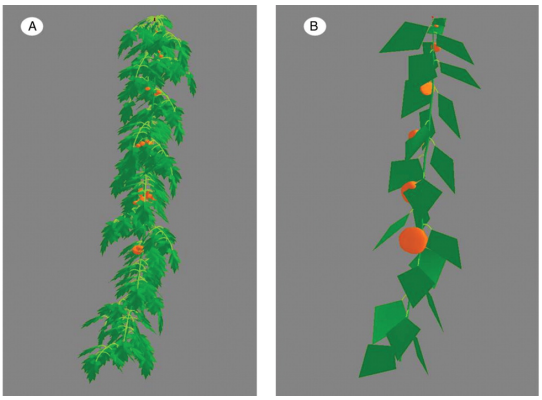
TABLE 2.1: Plant and crop models reviewed in this chapter categorized in model types

Category	Name	Plant Part	Programming Language	Modelling Framework	3D Output	Reference
FSPM	AMAPSim	Shoot	C++	AMAPstudio	Yes	[40]
	CPlantBox	Whole Plant	Python	Stand Alone - CRootBox	Yes	[41]
	GreenLab	Shoot	Matlab, Java, C++, Scilab	GreenLab	Yes	[42]
	L-PEACH	Whole Plant	L+C	L-Studio	Yes	[43]
	Helios	Shoot	C++	Stand Alone	Yes	[44]
	GroIMP	Shoot	XL-System, Java	Stand Alone - GROGRA	Yes	[45]
	V-Mango	Shoot	Python, L-Py, R	OpenAlea	Yes	[28]
	QualiTree	Shoot	UML	Stand Alone	Partial	[46]
FPM	ARCHIMED	Shoot	Java	AMAPStudio	Yes	[47]
	RATP	Shoot	Python, F90	OpenAlea	Partial	[48]
	Tomato	Shoot	Java	GroIMP	Yes	[49]
SPM	OpenAlea	Shoot	Python	OpenAlea	Yes	[50]
	MAppleT	Shoot	C++, Python, L-Py	OpenAlea	Yes	[51]
	Top-vine	Shoot	Python	OpenAlea	Yes	[52]

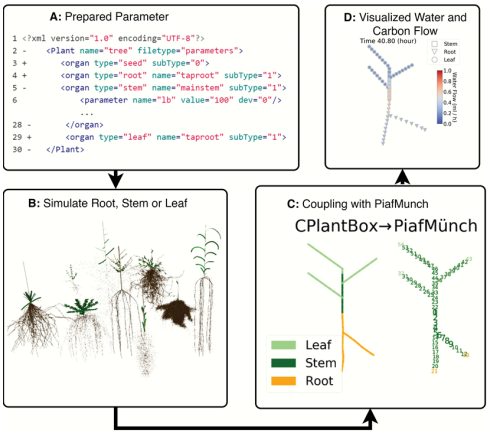
2.2.1 Functional–Structural Plant Models (FSPM)

Barczi et al. proposed in 2008 a structural whole-plant architecture simulator called *AMAPsim* [40]. The main goal of *AMAPsim* model is to combine architectural and physiological models, including both a general botanic framework and plug-in functions. The model and software, written in C++ language, is mostly based on those proposed by Bouchon et al. [53]. At its core, *AMAPsim* is a model designed to reconstruct a 3D plant structure (SPM) using a set of botanical parameters and concepts as growth engine. It has been classified in this work as an FSPM since, thanks to its modular structure, it is possible to use plug-ins for modeling the functional part of plants, becoming in all respects an FSPM. *AMAPsim* describes the plant as a set of hierarchical levels starting from the axis, annual shoot, growing unit, zone, and internode. The model considers the bud as a potential growth point. Once the topology is determined, the plant components are placed in the 3D space according to the method proposed by Jaeger [54] by associating geometric shapes to each plant component, e.g., a cylinder to the internode and polygonal surfaces to the leaves. These 3D components are placed in the environment according to geometric rules. The curvature of the branches by their own weight is modeled according to beam theory. The initial direction of the axes is defined according to phyllotaxis, and the initial length and diameter vary over time according to proportion rules. In its basic form as SPM, *AMAPsim* does not consider effects such as heliotropism or epitony. However, due to the possibility of using plug-ins, these features were included later as external module. The authors showed an example of their FSPM based on the *AMAPsim* core, using tomato plants. Figure 2.1a shows the results of this simulation: on the left (A) is shown a complete shape with detailed leaves; on the right (B), a simplified shape with leaves is approximated to an equivalent area. As shown in Figure 2.1a, the 3D output can be defined by to the computational power and the wanted precision; however, it is not a real-time 3D rendering, and it is not possible to modify environmental variables runtime except at the end of every time-step of simulation. The modularity of the *AMAPsim* model, as well as the possibility of being able to use it without knowing programming languages, allows its use by non-programmers simply by calibrating the input parameters.

In 2020, Zhou et al. [41] introduced *CPlantBox*, a whole-plant functional–structural modeling framework. The authors proposed a model that can simulate both root and plant shoot as full plant structure and as a single topological network of organs. The model is written in Python language, and the input parameter file are XML-based to increase flexibility. As output, *CPlantbox* generates a one-dimensional graph with 3D topological coordinates. The nodes of this output graph have properties such as 3D coordinates, radius, organ type, water potential, and others. The workflow of the model is shown in Figure 2.1b. The model is capable of generating a 3D plant architecture for both the hypogeal and epigeal parts. For the epigeal part, *CPlantBox* is able to generate different patterns of leaf orientation, depending on the parameters set. The different structures are generated by changing the input parameters. *CPlantBox* is also capable of calculating the impact on the plant of a heterogeneous environment. In this case, the input parameters are assigned manually and punctually and cannot be varied during the run-time of the simulation. The novelty introduced by *CPlantBox* consists in simulating as a single network both the roots and the shoot and the possibility to couple it to water and carbon flow models, making it a full FSPM. While having a 3D output of the plant architecture, we believe that an improvement of this aspect can bring several benefits to the model itself: a better 3D representation of the leaves and of the plant structure itself, as well as an improved calculation of the ambient light input, can lead to a more accurate calculation of the photosynthesis process, improving the water and carbon flux model data. In addition, improved 3D structure can be useful for calculating competition between adjacent plants for both light and resources.



(A) 3D output of *AMAPsim model*: tomato plant modeled with complete shape (a) and simplified leaves shape (b)



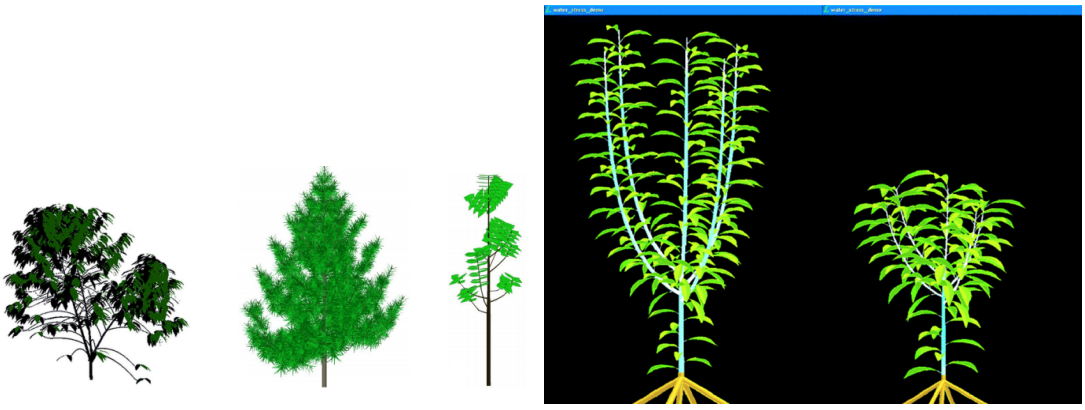
(B) Major steps, as flow chart, in *CPlantBox* framework coupled with *PiafMunch*

FIGURE 2.1: (**A**) *AMAPsim* model 3D output (Barczi et al. [40], with permission from Oxford University Press, 2021 license 5200260057623) and (**B**) working flow chart in *CPlantBox* model (Zhou et al. [41] 2020. Published by Oxford University Press on behalf of the Annals of Botany Company. Distributed under the terms of the Creative Commons Attribution License)

The *GreenLab* mathematical model proposed by de Reffye et al. [42] is based on the architectural foundations of trees, identifying the basic components of plant development. A key assumption is that axes of the same type are duplicated at different stages of development. *GreenLab* is an FSPM that integrates the functional and structural part of metabolic processes with a phytomere-level structure. The model has been developed over the years in different development environments and programming languages. The development of the plant structure and its formalism is based on a double-scale stochastic automaton. The 3D output provides the entire topological basis, having a recursive system of equations as its basis (Figure 2.2a). In addition to the graph visualization, *GreenLab* can also produce simple 3D structures (Figure 2.2a), which are purely geometric since photosynthesis is not contemplated. As stated by de Reffye et al. [42], compared to FSPM and crop models, *GreenLab* falls "in the middle", as a voluntarily simplified model both from the geometric and structural point of view and from the physiological one. It is a "source–sink solver": a tool capable of solving source–sink dynamics during growth. Moreover, as stated by the authors, unlike other FSPMs, *GreenLab* does not provide a fine description of physiological processes. Its strength lies in the fact that it can be used both as a solver of source–sink relationships and to model individual plants up to stand level. In addition to further developments that are underway by the authors, the possibility of integrating *GreenLab* with a more performant and precise 3D geometric structure could be a further step towards improving the model.

The L-PEACH model proposed by Allen et al. [43] is based on the PEACH model by Grossman and DeJong [55] which models carbohydrate source–sink dynamics of vegetative growth of fruit trees. While making it possible to avoid the use of empirical allocation coefficients, functional balance rules, or fixed allometric relationships, the PEACH model completely ignored the relationship between architecture and carbon allocation. In addition, it was not possible to evaluate differences in organ size based on its position on the structure. The integration with *L-systems* allowed these limitations to be overcome. L-PEACH was written in L+C language and implemented in *L-studio* [56]. The plant model is also sensitive to variation in water concentration, affecting carbohydrate production and flux. Figure 2.2b shows how the simulated plant reacts to different concentrations of water: the tree on the left has been simulated under conditions of full irrigation and the one on the right with mild water stress during growth. The *L-systems* used in the L-PEACH model facilitates the integration of individual organ models into a branched growth structure as well as dynamically updating the system of equations governing carbon partitioning. It also allows the interface between the plant model itself and the ambient light partitioning model, as well as the water concentration model. L-PEACH attempts to overcome some of the limitations that tree growth models based on carbon partitioning have had, as reviewed by [57]. However, the authors declare that more testing is needed to have sufficient data to fully calibrate the model [43]. The flexibility given

by the *L-systems* allows to adapt the studied model to different case studies. Integration between the two fields is promising and possible, as discussed in the following sections.



(A) Various geometrical structures of trees generated by *GreenLab* model

(B) Two peach trees structures with different water intake made by *L-PEACH* model

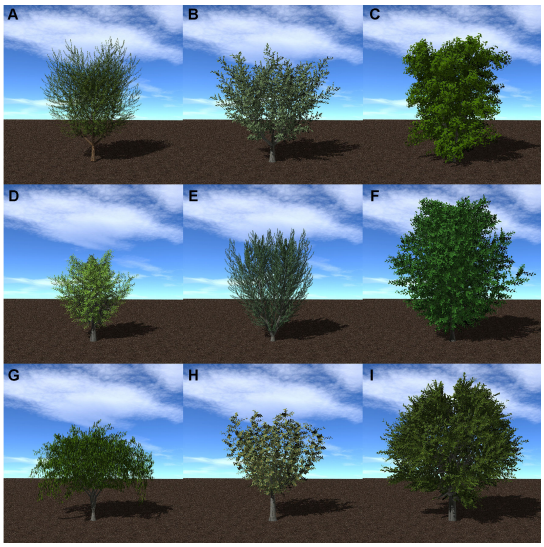
FIGURE 2.2: The *GreenLab* (**A**) (de Reffye et al. [42], with permission from Oxford University Press, 2021 license 5200260511673) and *L-PEACH* (**B**) (Allen et al. [43], with permission from John Wiley and Sons, 2021 license 5200251150244.) models with their 3D structure output

An integration between FSPM models and 3D rendering is introduced by Bailey in 2019 [44] as a framework for plant and environmental modeling called *Helios*. The author pointed out some important trade-offs in plant systems models and proposed *Helios* as a different framework in which the mentioned trade-offs are prioritized. The complexity of the model is one of these trade-offs: simple models such as the "big leaf" model proposed by Sinclair et al., Raupach and Finnigan, Friend [58, 59, 60] can lead to very low computational effort but introduce inherent errors and biases into the model approximation. On the other hand, more precise models that simulate each individual leaf [61, 50] are more accurate but computationally heavier. Another trade-off pointed out by Bailey is the choice between ease of use and flexibility. Increasing the ability of the end user to have more control over the configuration of the software inevitably results in a decrease in the ease of use of the model framework. By automating some of the tedious and technical tasks, developers can increase ease of use but also create multiple limitations for advanced users. An example is given by the non-sequentiality of processes: the simulation of the photosynthesis, as an example, is based on cyclical processes linked between them. Coupling these processes into a model requires iterations, which requires flexibility when incorporated into a generalized modeling framework. This problem was also discussed by Pradal et al. in the development of the *OpenALEA* model [50], where some of the flexibility was sacrificed in the name of ease of use for the end user. The author used C++ language as more efficient and flexible sacrificing some of the usability [44, 56]. Sacrificing ease of use, the model is formulated to allow maximum

user control over plugins (submodels), data flow, and execution. *Helios* is capable of using complex biophysical models, utilizing the parallel computing capability of GPUs to reduce computational weight. The core of the model is represented by a class called Context, which manages the data and geometry of the model. The output of the model is a 3D representation of the plant that can be exported as standard OBJ or PLY files, coupled with XML files for simulation data. This way, coupling of geometric data with other models such as *GroIMP* or *OpenAlea* is possible. The generation of 3D tree structure in *Helios* is possible thanks to the procedural tree generation submodel. This plug-in allows to create semirandom tree geometries through the algorithm proposed by Weber and Penn [62]. The woody architecture of trees is generated by recursive sets of branching levels with a random perturbation. The procedural tree generation plug-in comes in *Helios* with predefined tree geometries as shown in Figure 2.3a. The final tree is similar to ones generated with *L-systems* approach. Although the *Helios* model has the ability to procedurally generate the 3D structure of the tree, this function appears limited, having a defined number of trees (predefined tree geometries). There is no possibility to generalize the procedure, as in the other plug-ins, to generate the architecture according to a mathematical biological model; additionally, it is not in real-time. We believe that an improvement in the tree structure generation part could improve the calculation of these submodel parameters by allowing the user to appreciate any differences caused by the different architectures generated, differences in light interception, competition between nearby trees, and more.

The concept of *relational growth grammars* (RGG) proposed by Kurth et al. [63] features a combination of L-systems and graph structures, using a generic programming language. This concept was introduced to make the L-system formalism more suitable for FSPMs since a plant is not the sum of individual parts and shows graph structures that cannot be represented with L-systems which are a linear string of symbols. The formalization of these concepts is exactly the concept of RGG, and its implementation is the XL programming language. The linear sequence of symbols of the L-systems is replaced by a real graph having a series of connected nodes and edges. Typical FSPM relationships can be represented by the graphs and their connections between nodes. The graphs themselves can be transformed by applying rules, such as the rule-based paradigm of L-systems. Kniemeyer et al. [45] put together *relational growth grammars* in an interactive modeling platform, based on the XL language, called *GroIMP*, with the aim of making the concepts of relational growth grammars usable to the end user. *GroIMP* has a module for modeling 3D parts, using geometric classes that include primitive shapes such as spheres, cones, boxes, splines, surfaces, and lights, as well as a library of materials applicable to shapes and surfaces. The generated 3D structure is rendered using the POV-Ray engine [64]. On the 3D structure, the user can apply some modifications such as the deletion of parts by interacting directly on the shape. The implementation of relational growth grammars in *GroIMP* was done via the XL programming language. It

is defined as an extension of the Java language and allows to maintain the syntax of the L-system rules. The language also allows queries and peforms mathematical operations on data sets. All of this allows an implementation of the process-based part of the plant models and links it with the structure. The structure visualization is shown in Figure 2.3b with a basic example of a maple tree. RGG is not the only extension of the L-system formalism: in the literature, it is possible to find several extensions that take different directions, such as context-sensitive ones that allow the modeling of local transport [56], global-sensitive ones that take into account the global environment by interfacing the L-system with an external program [65], and those of implementation of different programming languages, such as L+C that adds the rules of L-systems to the programming language C++ [66]. These extentions remain in the framework of the L-system, and *relational growth grammars* extend the formalism of L-systems towards the graph grammars to model the complex FSPM in a natural way. Kniemeyer et al. [45] argue that the advantages of RGG are that it can store tree- and graph-like structures as data instead of a sequence of symbols that is inadequate for FSPM modeling of graph-like structures. The representation of structures is also immediate, avoiding the interpretation of turtle steps as an algorithm, simplifying the interpretation of global and local rules. Moreover, the use of a generic programming language is useful to implement the process-based part of FSPM. L-systems have this kind of integration, for example with the already mentioned L+C language with the L-PEACH model aforementioned [43], but does not define any syntactic support for accessing and calculating on the entire structure. The calculated graph is exportable to be modeled in 3D software such as CINEMA 4D or Maya. However, the 3D structure is only a graphical representation and not a functional object for the model itself. Such a use could be in our opinion an important step to further extend the model. The authors also pointed out that *GroIMP* cannot solve systems of differential equations that are often the basis of FSPMs. *GroIMP* has been implemented for modeling different plants and performing functional–structural calculations, as shown by Zhu et al. in *GrapeVineXL* [67].



(A) Different 3D trees generated with procedural tree generation plug-in in *Helios*

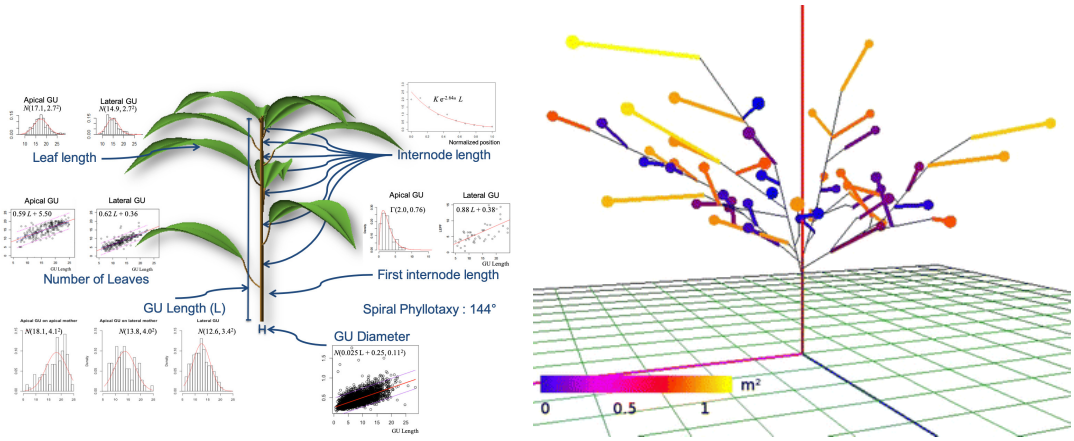
(B) *GroIMP* application on a 3D generated maple tree

FIGURE 2.3: *Helios* (**A**) model outputs (Copyright © 2019 Bailey [44], distributed under the terms of the Creative Commons Attribution License) and *GroIMP* (**B**) generated maple tree (Reproduced with Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Netherlands from Kniemeyer et al. [45])

An example of the application of FSPM as an agronomic tool for crop practice design is *V-MANGO* proposed by Boudon et al. [28]. The model is based on a representative FSPM of the mango tree (*Mangifera indica* L.) that reproduces the vegetative and reproductive development of the tree and their interactions. The main goal of the authors has been to create an integrative FSPM model of mango tree development and fruit production to demonstrate that an FSPM can be used to formalize the development of the complex architecture of mango trees also phenologically, to show how the introduction of some temporal and structural factors can simulate the complex interactions between vegetative and reproductive growth, to create a first basis for a model that can be used for the design of cultural practices. *V-MANGO* is based on the architecture model proposed by Hallè el al. [68], described as a series of growth units, inflorescences and fruits organized as an arborescent structure. Growth units are composed of a series of internodes and leaves organized in a spiral pattern. Boudon el al. [28] modeled several sub-models to integrate these different scales and combined them consistently. A plant architecture model was first defined simulating the visual part of the different entities (growth units and inflorescence) by decomposing morphogenesis into elementary stochastic processes. Growth unit geometry is defined by several variables such as axis length and diameter, number of leaves, leaf length, width and area, internode length, and phyllotaxis (see figure 2.4a). Using all of these parameters, the authors have been able to accurately represent the geometry and location of each individual growth unit. All these growth

unit, fruit, and inflorescence models have been assembled by the authors into a complete architecture using an L-Py [69] module of the *OpenaAlea* [50] platform, using L-system formalism with the Python language. The visualization of the structure is then entrusted to the L-system formalism and rules. Although *V-MANGO* is a complete FPSM, which models both the structural and functional part of the mango tree and has as output a 3D structure of the tree, it has further integrations and updates to be implemented, according to the authors. The 3D structure generated by *V-MANGO* lacks a module to calculate the amount of intercepted light, and the introduction of this module would allow a more accurate calculation of photosynthesis in fruit production. Finally, the introduction of modules that take into account agronomic practices, such as pruning, would allow the *V-MANGO* model to be used as a tool for precision agriculture.

A more generic model aimed at modeling fruit trees is *QualiTree*, presented by Lescourret et al. [46]. The authors presented *QualiTree* as a model that can simulate the effects of different agronomic practices on fruit tree development, particularly on the quality of fruits produced. The proposed model is highly targeted towards the study and modeling of the quality of the fruits produced, focusing on some functional aspects of their growth. The modeling of the tree was done considering only the characteristic traits of the quality of the fruits produced (Figure 2.4b), leaving the other parts of the tree separate as different compartments. From a level of detail point of view, *QualiTree* falls in the middle between compartmental models and those that use the description of individual organs to represent the architecture of the tree, such as the other FSPMs described in this study. The hierarchical structure has been developed with UML language which, through a representation based on object/class programming paradigm, allows the hierarchical modeling proposed. The functions (mathematical operations) encoded in the model have been classified as specialized models and generalized models. According to the authors, specialized models include all mathematical equations describing carbon allocation according to source–sink systems and their balances, while generalized models include those describing leaf photosynthesis, carbon assimilation by fruits, mobility of carbon reserves, daily growth requirements and others. Some of the improvements proposed by the authors for *QualiTree* are the extension of the fruit quality parameters, the amplification of the effects on the cultivation practices model, but above all the integration with *QualiTree* of light interception models for a more precise calculation of photosynthesis. In this regard, we believe that the implementation of a generic model such as *QualiTree* with a 3D rendering system of the structure, thus overcoming the fact that the structure is already defined as input data, can improve the calculation of light interception and consequently the calculation of photosynthesis and all processes related to it. The authors also propose to scale the model in the future from a tree scale to a tree group scale in addition to increasing the time scale from one growing season to multiple consequential and consecutive seasons.



(A) Mango tree structures, parameters, and model calculations in *V-Mango* FSPM

(B) Fruit tree simulated in *QualiTree* with representation of Fruits Units (FU—colored lines) and old wood (black lines)

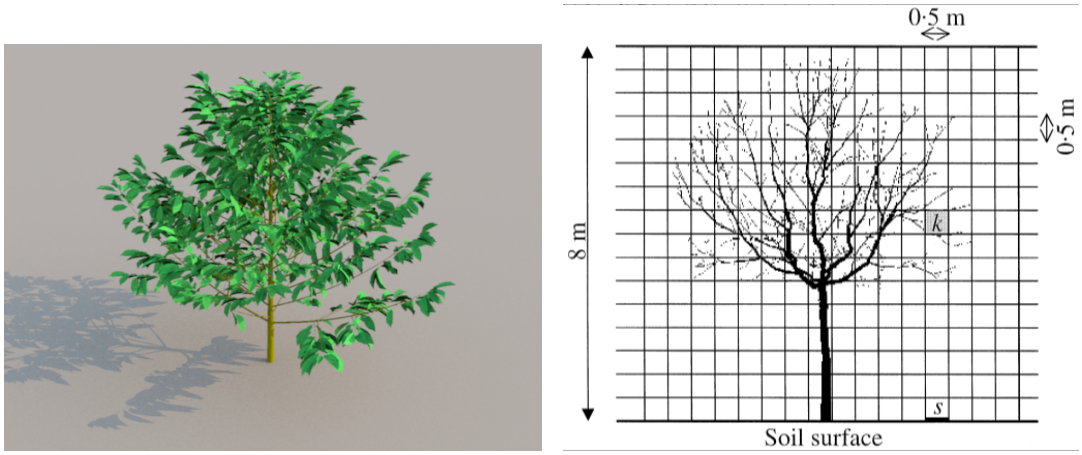
FIGURE 2.4: *V-Mango* (**A**) FSPB (Boudon et al. [28] 2020. Published by Oxford University Press on behalf of the Annals of Botany Company. Distributed under the terms of the Creative Commons Attribution License) and *QualiTree* (**B**) model output (Reproduced with permission from Lescourret et al. [46], published by Springer, 2011, license number 5198331325727)

2.2.2 Functional Plant Models (FPM)

The *ARCHIMED* model proposed by Dauzat et al. [47] integrates biophysical modules to model the complex interactions between leaf, plant, and plot scales. The authors' choice was to use a single-leaf scale for most processes such as photosynthesis, leaf radiation, and energy balances. The modules present in *ARCHIMED* allow the calculation of several biophysical processes such as the radiative balance of organs, their transpiration, and their carbon assimilation. The *ARCHIMED* platform includes two sub-platforms called *ARCHIMED-*φ and *ARCHIMED-*ψ, respectively specialized in biophysical modeling and in 3D plant growth modeling. The processes of photosynthesis and transpiration are controlled by stomatal conductance parameter which can limit photosynthesis by reducing carbon dioxide influx and indirectly modulates leaf temperature through evaporation. In order to be able to calculate all the parameters involved in the simulation, *ARCHIMED* presents a system of modeling of the light and its quantity that arrives on the structure (in particular on the leaves) as photosynthetic active radiation (PAR). The light input is not calculated in real time by the 3D environment, but it is a prior measurement. The internal behavior of the leaves is governed by the water potential calculated according to a sap flow model. *ARCHIMED* is capable of modeling interactions between neighboring plants in the same plot. Each plant modifies its own environment either due to its proximity to another plant or due to external elements such as wind. The model used by *ARCHIMED* to calculate the light flux in the plot is the MMR (MirMusc-RadBal) model as described in Dauzat et al. [70]. As shown in Figure 2.5a, through the *XPlo* component of the *AMAPstudio* platform, it is possible to generate

and render 3D tree structures from *ARCHIMED* model data. In the Figure 2.5a, a coffee tree was generated and rendered. Through *AMAPstudio*, it is possible to modify the architecture, evaluate data, and visualize the structure. The modularity of the simulation platform allows the future integration of other models to extend and refine the system, as proposed by the authors themselves.

The functional tree model presented by Sinoquet et al. in 2001 [48] called *RATP* (Radiation Absorption, Transpiration, Photosyntesis) allows the simulation of the spatial distribution of radiation and leaf gas exchange within the vegetation as a function of its structure and microclimate. The model has as input tree geometry, microclimate variables, and properties of tree components. In the research proposed by Sinoquet et al., the model was calibrated on a walnut tree on which all input parameters were measured. The *RATP* model does not allow a structure generation per se, but the structure itself is input data. *RATP* according to the authors can be used to study the allocation of resources between vegetative components and between organs in the same plant. The model does not simulate the microclimate inside the canopy except for radiation and wind speed in small scale according to empirical relationships. As proposed, *RATP* is a model used to study water and carbon exchanges at the interface between the plant and the atmosphere. It does not have the ability to generate a 3D model of the plant, and the input model of the structure is given by field measurements of real trees. Thus, there is no exchange of information between the virtual environment and the model with its parameters. Being a generalized FPM model, it would be possible to integrate it with 3D rendering engines to increase the accuracy of some calculations such as absorption and interception of solar radiation, leaf distribution on the canopy structure, leaf irradiance measurement and so on, relying on a precise architecture of synthetic tree.



(A) (B)

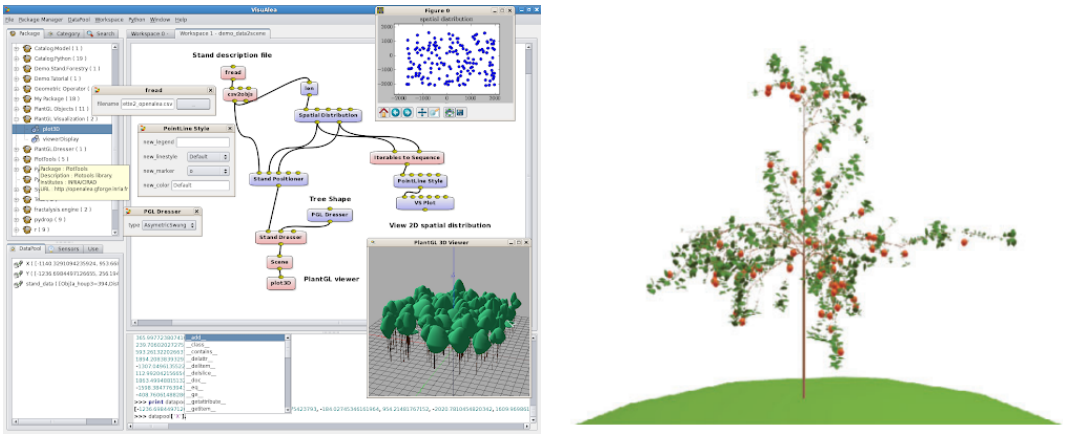
FIGURE 2.5: (**a**) Coffee tree simulated in *ARCHIMED* with XPlo AMAPstudio's module and Sunflow integrated renderer; (**b**) walnut tree structure with division in 3D cells made by *RATP* model, Sinoquet et al. [48], with permission from John Wiley and Sons, 2021 license 5200250775819.

FPM models can also be used on non-tree plants. The applications in the literature are numerous, such as modeling the functionality of the tomato plant. In particular, some studies have been performed using a virtual plant model to optimize certain growth conditions. For example, De Visser et al. [71] used in their study a 3D model of tomato plant coupled to a ray-tracing model of lighting for the optimization of the light itself showing that illumination efficiency increases when the lamp light is directed at most to leaves that have a high photosynthetic potential. Dieleman et al. [72] have instead demonstrated how, by conducting tests in a 3D virtual environment, dynamic light spectra may offer perspectives to increase growth and production in high-value production systems such as greenhouse horticulture and vertical farming.

2.2.3 Structural Plant Models (SPM)

The main feature offered by the *OpenALEA* model is to provide researchers with a visual and interactive interface to an FSPM application. *OpenALEA* due to these features has been used as the basis for several models, some of which are mentioned in this paper [28, 51, 73, 74]. The visual programming environment was conceived by the authors as a way to integrate and connect the model rather than model feedback and retroaction, which can still be done using specific dataflow nodes and biophysical solvers (Figure 2.6a). The intricate interactions between plant functions, such as hormonal control, resource partitioning, and water flow, are partially modeled and require the definition and sharing of generic data structures representing different plant processes at different scales and in parallel. Due to the fact that *OpenALEA* is a general purpose model, the authors have chosen to develop it in an open-source, object-oriented, platform-independent language with an easy-to-learn syntax such as Python [75]. *OpenALEA* is based on the "component framework" principles introduced by Councill and Heineman [76] that allow the dynamic combination of independent pieces of software into a workflow. The authors have also developed a visual programming environment, *VisuAlea*, that allows scientists to create complex models without having to learn a programming language. In *VisuAlea*, it is possible to graphically combine the different modules, nodes, and components provided by the *OpenALEA* library and then run the model. Thanks to its modularity, *OpenALEA* presents basic components already integrated for the modeling and analysis of plant architecture, geomentric plant modeling, ecophysiological processes, and meristem modeling and its simulation. *VPlants* is the successor of *AMAPmod* and is a package for the modeling and analysis of the plant architecture, while for the modeling of the geometry, there is the *PlantGL* graphics library that uses the procedural rendering method proposed by Weber and Penn [62]. As described, the integration of modules in the *OpenALEA* model allows the creation of 3D structures of plants aimed both at the visualization of the model and at a functional use such as for the calculation of light and rain interception. The procedural methods of shape generation used in *OpenALEA* will be described in Section 2.3.

An example of *OpenALEA* application is given by the *MAppleT* model presented by Costes et al. [51], which aims to model the apple tree using biomechanical properties and mixed stochastic models. The integration of topology and geometry models in a single simulation in order to make the architecture of an apple tree emerge from the interactions between the processes was achieved by the authors using L-systems. At the time of its publication, *MAppleT*, together with *L-PEACH* already described in this paper [43], was one of the first approaches to the simulation of fruit trees with development over the years and responsive to gravity. The 3D output turns out to be only visual, used by the authors as a comparison with data measured on real trees for model validation. It is not used as input to the biological model, e.g., for calculating light interception, as well as in other models described in this work.



(A) (B)

FIGURE 2.6: (**a**) Overview of *OpenALEA* software interface with dataflow and interconnectind nodes. On the lower right, the 3D viewer is shown. Screenshot from software. Main research by Pradal et al. [50]. (**b**) A five-year-old apple tree modeled with *MAppleT*. Adapted from Costes et al. [51] with permission from CSIRO Publishing, 2021 license 1165429-1

FSPM, SPM, and FPM models are also used to evaluate how the structure of a plant can modify its characteristics, understood as the ability to intercept light, fruit distribution, production, gas exchange, water transport, and so on. These solutions are also adopted for high-revenue species such as fruit trees and vineyards. In the case of the vineyard, several studies have used these models to simulate the plant in its functions and structure. An example is given by *Top-Vine*, proposed in 2005 by Louarn et al. [52]. This model is proposed as a topiary approach based on an architectural model to simulate vine canopy structure with a 3D distribution of leaf area for contrasted vine training systems. Leaf area is an input to the model that operates at the shoot level and considers each shoot to have the same leaf area. The volume is refined by the inputs which are field measurements of real vines using a 3D electromagnetic digitizer. *Top-Vine* has been implemented on the *ALEA/OpenALEA* platform. The authors noted a close relationship between the field data and the model

results at both the canopy and organ level, allowing, once the model is calibrated, to be able to generate a structure at any desired time-step and allowing a light interception calculation. Thus, as in the model proposed by Louarn et al., other authors have also used FSPM, SPM, and FPM models to virtually recreate the vine. Models such as the one proposed by Zhu et al. [67], thanks to a 3D reconstruction of the plant, allow the calculation of gas exchange, coupling it to the dynamics of water transport.

2.2.4 Critical Comparison of Modeling Systems in Agronomical Applications

The development of FSPMs has matured significantly over the past 20 years and has already provided significant results at different levels of organization and in different fields of plant science. FSPM models are now at the core of fundamental questions in plant biology and predictive ecology. The models studied in this work exhibit specialization on a single plant or are generic frameworks on which specific models have been developed. In particular, FSPMs present a more holistic modeling, often of the whole plant or part of it (shoot or root), both the functional and the structural part. The variety of model applications has also allowed the development of SPM and FPM models, as a branch of FSPM models, which are specialized in a particular aspect: structural only or functional only, by choice of the modeler and for the aims of the development. Table 2.2 shows the differences between the various models analyzed in terms of model aims, agronomic applications, the presence or absence of plant–plant and plant–soil interactions, and the organs and processes modeled.

TABLE 2.2: Overview of reviewed models characteristics, aims and agronomic applications

Category	Name	Plant Organs						Plant-plant Interaction	Plant-Soil Interaction	Modules / Processes				Aim of the model	Agronomic applications
		Bud	Stems	Fruits	Leaves	Flowers	Roots			Carbon allocation	Photosynthesis	Water allocation	Other		
FSPM	AmapSim	x	x	x	x			Yes (external module)	No	x	x			Plant architecture; biomass/fruit production; carbon balance; light interception; stand design; pruning effects	Allometric models calibration; density and planting pattern effect on architecture; quantitative analysis of branching patterns; tree response to wind [77, 78, 79, 80, 81, 82, 83]
	CPlantBox	x					x	No	Yes	x				Carbon and water balance	In-field phenotyping [84, 85, 86]
	GreenLab	x	x		x	x		Yes (static input)	No	x	x	x		Biomass production; optimization and control of farming systems	Phenotyping; plant breeding; [42, 87, 88, 89, 90, 91] crop management
	L-PEACH	x	x	x	x		x	No	No	x	x		Source/sink strength	Horticulture; carbon accumulation and fluxes ODEs solutions	Crop load effects; responses to canopy management [92, 93]
	Helios				x			Yes (Light competition only)	No		x		Energy balance; stomatal conductance; radiation model	Plant architecture; light interception	Vineyard management; grapevine disorder estimation; [94, 95, 96, 97] 3D phenotyping
	GroIMP	x	x		x			Yes (Light competition only)	No	x	x		Radiation model	Tree-structure generation; horticulture; carbon balance	Genotypic and phenotypic characterization; visualization of field measured plant architectures [98, 99, 100, 101, 102, 103]
	V-Mango			x		x		No	No	x (to fruits)		x		Fruit production; tree phenology	Crop management [104, 105, 106]
	QualiTree	x	x	x				No	No	x	x			Fruit quality	Crop management [46, 107]
FPM	ARCHIMED	x			x			Yes	Yes (static water potential input)		x		Radiation balance; energy balance; sap flow	Ecophysiological processes modeling	Crop management [47, 108, 109]
	RATP	x	x	x				No	Yes (radiation exchange only)		x		Radiation absorption; transpiration	Light interception; leaf-gas exchanges	Test of plant functions; [48] plant reaction to drought
	Tomato		x		x			Yes (Light competition only)	No	x	x		Radiation model	Plant architecture; horticulture; carbon balance	Genotypic and phenotypic characterization; [49] visualization of field measured plant architectures
SPM	OpenAlea	x	x		x			Yes	No	x		x	Radiation interception; transpiration; rainfall interception and distribution	Plant architecture analysis; ecophysiological processes; meristem modelling	Biomechanics; [50] crop management
	MAppleT	x	x		x			Yes	No	x			Gravitrophism; light interception	Plant architecture	Biomechanics; [51] crop management
	Top-vine		x		x			Yes	No	x			Radiative balance; light interception	Plant architecture	Biomechanics; crop management; [52, 110] evaluation of vine training systems

As shown in Table 2.2, among the various models reviewed, there is a large heterogeneity between the processes considered, the organs modeled, and the agronomic applications. *AMAPsim* uses the processes of carbon allocation and photosynthesis, modeling buds, stems, fruits, and leaves. The carbon allocation is managed by external modules, which is possible thanks to the possibility of connection with external modules specialized in particular functions. In the case of *AMAPsim*, the external module of carbon allocation is a specialization of the one already used in *GreenLab* [111, 42] that calculates the storage of biomass, photosynthesis and biomass allocation for each time unit according to current climate and current plant shape, the length according to allometry on biomass, and the diameter according to allometry on biomass. The allocated carbon is used to determine the size of the organs, which in the case of *AMAPsim* are buds, stems, fruits, and leaves. *CPlantBox*, unlike *AMAPsim*, considers the plant as a single connected network of organs and models the allocation of water and carbon not as a redistribution of the data calculated according to the topology but as a flow generated by the potential difference between the various organs of the plant according to the Münch theory [112]. In this way, the model is able to simulate plant and roots as a single network, generalizing the parameters so that they can be adapted to field measurements. The difference in modeling approach and modeled organs leads to a number of different agronomic applications. Applications of *AMAPsim* have been found in the literature in several areas such as studying the response of trees to wind, as shown by Sellier et al. [79], quantitative analysis of branching patterns [78], and how planting density and density can affect plant architecture [82]. These applications have been made possible, as described in the previous section, by the integration of specialized external modules into *AMAPsim*. The different modeling approach of *CPlantBox* has allowed its application in other agronomic fields such as in-field phenotyping. Magistri et al. [84] used *CPlantBox* to rely on synthetically generated models instead of scanning real plants in a controlled environment and use them to measure phenotypic traits in an automated fashion without intensive manual work. *GreenLab*, as described in subsection 2.2.1, is presented as a model that starts from a concise mathematical representation at the phytomer level of metabolic processes to describe plant growth and development at the individual level up to the stand level [42]. Carbon allocation was assessed by computing, according to the source–sink concept, the strength of source organs and sink organs, calculating the plant demand as the sum of all active sink organs at a given time t. The root system is considered as a simple, single compartment, focusing only on the epigeal structure. The equation governing the model has the form of a discrete dynamic system, as a function of time, set parameters, allocated biomass, and environmental conditions over time. The development of the botanical structure was developed in *GreenLab* by describing the physiological rules of each new phytomer produced, using a graph notation approach. The structures produced can be simple or complex depending on the characteristics of the meristems. Random events that alter the development of the plant architecture

are introduced into the model as stochastic functions that modify the automaton transition rules. The validation of the model was found to be satisfactory for about 20 plant species (herbaceous, shrubs, and trees) showing stability of the estimated parameters in different climatic conditions. Being a deliberately simplified generic model, both from the point of view of ecophysiology and of representation of the structure, the agronomic applications are different, even with the necessary approximations. Feng et al. [89] used *GreenLab* to calculate the 3D structure of maize after evaluating and calculating the various parameters at a given planting density, showing good adherence to the measured data. In plant breeding, Letort et al. [90] considered the growth parameters, in first approximation, as invariants leading to an efficient selection in the search of ideotypes. Moving from individual plant to stand, the approximation of parameters increases as parameters are partitioned using a production area parameter, a function of density. Baey et al. [91] used this approach to estimate stand biomass as a function of density, showing comparable performance to other specific models. *GreenLab*, according to their authors, has a modeling rigidity given the fact that feedback between growth and development is ignored in addition to the lack of mortality models, things that will be supported in future developments. Carbon allocation modeling has also been used in the literature to estimate fruit production and quality, as shown in *L-PEACH*, *V-Mango*, and *QualiTree* among the models analyzed in this work. *MAppleT* also models carbon allocation, but in this case it is not targeted to fruit production and quality but to plant architecture by simulating gravitrophism and light interception. Returning to the models for estimating the production and quality of the fruit, *L-PEACH* is proposed as a model aimed at studying the effects of crop load, in this case for the peach tree. The model, like *GreenLab*, estimates carbon allocation through source–sink interactions and the calculation of their potential. *L-PEACH* simulates the effects of management and genetic and environmental factors that may affect the plant through complex interactions between plant organs, obtained by manipulating and adjusting parameters such as fruit number, fruit behavior, and the stems' storage capacity. The model is structured by modules, each for individual plant organs. Each individual organ can be evaluated as a source or sink of carbohydrates. The physiological parameters used were treated as electrical counterparts in that the calculation of carbohydrate storage, flux, and partitioning (amount (mass) of carbohydrate, carbohydrate concentration, carbohydrate flux, rate of photosynthesis, source/sink strength, resistance to concentration-driven flow) was done using equations studied in principle for the calculation of linear electrical circuits. The model is also interfaced with a method for calculating light distribution using a quasi-Monte Carlo method. The proposed model works well in a variety of agronomic applications such as canopy management. As shown by Lopez et al. [93], *L-PEACH* has been used to guide experimental research by helping to identify or develop quantitative hypotheses of potential yield-limiting processes that can be measured in the field, as well as the reaction of trees to external manipulations such as pruning. *L-PEACH* is

a good example of how architectural growth, carbohydrate assimilation and partitioning, and organ growth can simulate tree growth and physiology to provide integrated understanding of the environmental physiology of trees, in this case peach trees [93]. While also modeling the allocation of carbon in fruit trees, *QualiTree* has a different approach from *L-PEACH* because it does not generate the structure of the tree with architecture algorithms but needs data for initialization of the model that are the numbers of leafy shoots and fruits for each FU (Fruiting Unit). The model, after getting also as inputs the environmental parameters such as temperature, humidity, and PPFD for photosynthesis calculation, proceeds to the estimation of fruit quality. Agronomic practices influence the status of various classes such as winter or summer pruning. Pruning changes the shape to the structure of the tree, and *QualiTree* takes this into account by recalculating the source–sink balance and carbon exchange depending on the number of fruiting units and the distances between them. The main application of this model is crop management. Miras-Avalos et al. [107] used *QualiTree* to model peach tree reactions to a pest attack in different domains: on different cultivars (Alexandra or Suncrest), thinning intensity, and presence or absence of pest attack, showing a difference in the results of the fruit quality calculation and offering satisfactory results. The model is limited only to the size, quality, and yield of fruits and needs to be expanded with other quality parameters and other modules such as the estimation of water absorption (to simulate irrigation practices). Designed to be a specific model for estimating fruit quality and quantity, *V-Mango* uses carbon and water allocation processes for fruits only. The modeled organs are fruits and flowers along with what the authors call Growth Units (GU), modeled using empirical distributions and thermal time. Fruit growth was determined using an ecophysiological model that simulated carbon and water-related processes at the fruiting branch scale. The generated 3D models are used to accurately calculate light interception and require several variables: axis length and diameter; number of leaves; individual leaf length, width, and area; internode length; and phyllotaxy. The values of these variables were measured from experimental data. The fruit growth model simulated the daily increase of fresh mass of each individual fruit and was calculated by dividing it into two growth phases: the first corresponding to cell division and the second to cell expansion. The mass of the fruit is calculated every day as a combination of the masses calculated in the two phases of cell growth considering also the external environmental conditions such as temperature, humidity, and light conditions. Although it is a specialized model on a single type of fruit tree, *V-Mango* lends itself to several agronomic applications such as studying phenology and how, for example, the tree under certain conditions reacts to a pest attack, as shown by Boudon et al. [28]. *V-Mango*, despite being applied in several crop management applications, has not yet implemented the effects of agronomic practices such as pruning that is instead present in other models reviewed in this work (e.g., *L-PEACH* cited earlier). Another specialized model on a

fruit tree is *MAppleT*. Unlike the previous ones, although modeling the carbon allocation through the representation of the bud organs, stems, and leaves, being an SPM (Structural Plant Model), it is aimed only at the study of the structure of the tree (apple) influenced by gravity (gravitrophism). To do this, the authors have used stochastic topology models, on a Growth Unit (GU) scale, modeling their succession according to Markov chains. In order to estimate the effects of gravity on the structure, the authors also modeled the secondary growth of GUs and generated the shape of individual branches according to the biomechanical component of the model that estimates the bending and twisting of the branch itself. The generated 3D tree models were compared with empirical measurements on real trees, showing similarity between the real and modeled data. The effect of gravity on branches is also very similar to what the authors measured on real trees; however, the effect of gravity on morphogenesis, i.e., simulated gravitropism, is partially implemented. The authors note that there are discrepancies between the measured and model data of branch inclinations that are found to be incorrectly simulated. While *MAppleT* has limitations, it is the first model to simulate the effect of gravity on structure and has been used subsequently for crop management applications and biomechanics studies. For example, Han et al. [113] used *MAppleT* to estimate the efficiency of light interception as a function of tree geometric trait. A different approach was proposed by Bailey with the FSPM *Helios* model [44]. The proposed model does not simulate carbon allocation but focuses more on the photosynthetic, energy balance, radiation pattern, and stomatal conductance part. To do this, the author built the model by separating it into "contexts", "primitives" (defined as basic forms of individual plant elements), and separate, specialized "plug-ins" where the user has full control over parameters and processes. The plug-in (or module) for modeling radiation transfer was developed by Bailey in 2018 using a reverse ray-tracing method for modeling the net radiative flux in leaves [114]. The surface energy balance module was developed by considering the surface area of the single primitive and some parameters such as Stefan–Boltzmann constant, emissivity, and surface temperature, as well as external environmental parameters. The energy module depends on the estimated position of the sun based on Gueymard's [115] study. The stomatal conductance module is based on the studies of Buckley et al. [116] where stomatal conductance is given by a hyperbolic function of photosynthetically active photon flux density and local vapor pressure deficit. The module for photosynthesis is given by the studies of Johnson [117], and the mechanistic biochemical model of Farquhar et al. [118] for *C3* photosynthesis. One of the results that this model shows is that there are minimal differences in simulations of radiation absorption, transpiration, and photosynthesis through most of the day and in daily integrated distributions between isolated and dense canopy trees, raising questions about the representation of trees using simplified biophysical models. The *Helios* model was primarily used to model grapevine and vineyard management. As shown by Bahr et al. [97], *Helios* allows the estimation of sunburn damage to berry in the vineyard as a function of

leaf structure, density, and position, as well as vineyard architecture. Another use of the model was that proposed by Salter et al. [96] in phenotyping using a virtual chickpea plant as support for validation of a 3D reconstruction system using structure from motion (SFM) algorithms. Radiative processes have also been modeled in other works analyzed in this paper, such as the *GroIMP* model, which is proposed as a generic model by combining L-systems with a graph structure and uses an inverse ray tracing system: the scene consists of a series of light sources and objects that absorb (or reflect) radiation. The ray is traced from the source to the target, in an inverse mode to what happens in a classical ray tracing system. The amount of light calculated at each point is used to estimate photosynthesis as the amount of carbon assimilated into the leaves, given the calculated radiation and leaf area. The carbon thus assimilated is redistributed among the organs, which in the case of *GroIMP* are buds and stems, via the carbon allocation model proposed by De Reffye et al. [119]. Being a generic model, *GroIMP* has been used in several agronomic applications: Vermeiren et al. [120] studied how the shape of the leaves of a tomato plant affects light interception, Kang et al. [100] characterized genotypically and phenotypically sunflower plantations based on differences in water budget, Streit et al. [102] studied with *GroIMP* modeling how light interception changes when the geometry of Scots pine trees changes, and Kniemeyer et al. [45] studied how different carrot plants compete with each other by self-shadowing processes and how planting can be optimized to maximize yield. Another generic model that exploits the potential of radiation processes is *OpenALEA*. This model simulates, by using specialized single-function modules interconnected with each other, the organs buds, stems and leaves, combining radiation processes with carbon and water allocation processes. Specifically, to analyze branching patterns and architecture, *OpenALEA* relies on the models proposed by Durand et al. [121] and Guedon et al. [122] based on statistical models of Markov chains and change point detection. For the ecophysiological study, *OpenALEA* relies on the models already proposed by Chelle and Andrieu [123] and Sinoquet et al. [48] Specifically, the radiation is calculated according to a radiosity algorithm where two surfaces, one emitting and the other receiving, are linked according to an equation that takes into account their transmittance, reflectance, and relative geometric position (how much they are exposed in front of each other in terms of surface area). Photosynthesis and relative carbon gain are computed for sunlit and shaded leaves of each vegetation component in each 3D cell. Photosynthetic rates are simulated according to C3 theory, as already seen for *Helios*. This model was used without including the potential limitation arising from triose phosphate utilization. The agronomic applications of *OpenALEA* range from biomechanical studies on fruit trees to crop management of vines, as shown by Louarn et al. [52]. The analyzed models *RATP*, *Tomato*, and *Top-Vine* share the modeling structure with *OpenALEA*, using the same carbon and water allocation processes in addition to radiation absorption processes. While *RATP* models bud organs, stems, and fruits, *Top-Vine* and *Tomato* model only stems and leaves. Again, the agronomic

applications are different. As for *RATP*, the model was used for studies of plant response to drought. The geometry of the tree canopy was described as an array of 3D cells of defined size on the three axes. Each cell can contain different parts of the tree or be empty (Figure 2.5b), while the soil was discretized into a series of zones that exchange radiation with the vegetation. The plant properties are physical, such as angles of orientation of branches and leaves, and physiological (reactions to environmental conditions of the photosynthesis process if there are leaves). The model has as output the radiative balance, organ temperatures, carbon, and water balance in shaded and illuminated areas in each 3D cell. Ngao et al. [124] used *RATP* to study the variability of leaf temperature and stomatal conductance enhanced by drought in apple tree. With *Tomato*, instead, Sarlikioti et al. [49] studied how plant architecture affects light absorption and photosynthesis in tomato plants, using as a model for the photosynthesis process the one already analyzed and proposed by Farquhar et al. [118], while the light input module is given by the studies of Chelle and Andrieu [123] by calculating it with the inputs being the amount of light absorbed by each plant organ, the leaf transmittance and reflectance coefficients for the upper and lower side of the leaf, and the light from the light sources that were used to simulate the sky. The experiments were carried out in a virtual greenhouse at the variation of the 3D architecture of the plant. The *ARHIMED* model has a different approach in modeling the photosynthesis process: it is controlled together with transpiration by stomatal conductance parameter which can limit photosynthesis by reducing carbon dioxide influx and indirectly modulates leaf temperature through evaporation. Light exchange has been modeled by Vezy et al. [108] as a discretized model along the directions, alternative to ray tracing and radiosity models, while the internal behavior of the leaves is governed by the water potential calculated according to a sap flow model. Perez et al. [125] used *ARHIMED* to study architectural plasticity in response to planting density of oil palm. Dauzat et al. [47] used *ARHIMED* to study ecophysiological processes on 3D Virtual Stands.

In the review of these works, a fragmentary nature of the approaches emerged, both from the model design point of view and more in depth, such as the choice of programming languages. It is sufficient to note that, just taking into account the works reviewed in this review, there are five different frameworks and ten programming languages, in addition to the fact that each model has its own set of processes, modules, organs, and applications. Many reuse modules already extensively validated in previous years, such as the C3 module of photosynthesis, while some propose new approaches but remain to be validated with field data. A greater uniformity of the models can be achieved by improving the standard of integration of the models, without losing a limited generality, as discussed by Louarn and Song [4]. The fragmentary nature of the proposed solutions can also be seen in the 3D output of the studied models: all of them present a non real-time 3D rendering, with different techniques. Only a few models use the 3D model as a functional part of

the model itself, for example, to calculate the amount of light collected in photosynthesis processes. We believe that an improvement in this aspect can benefit the development of FSPM models, such as in the paper by Zhu et al. [17], in which the 3D architecture of plants in the field is derived from photography of real plants. Their study applies this method to assess light distribution throughout the growing season. The 3D outputs of FSPMs could also be involved in the development of deep learning methods used for automatic image analysis: the realistic representation of plant architecture that FSPMs can generate could serve as a novel way to provide inexpensive datasets of rendered images for a wide range of phenotypes [4, 126, 127].

2.3 3D Tree Rendering Techniques

As discussed in the previous sections, the analyzed models allow a 3D reconstruction of the results. The 3D model is generated using computer graphics techniques that can be generally classified in four categories: particle system and procedural methods, rule-based methods, and hybrid methods [16]. These methods are the most used and are based on the most recent and advanced techniques of 3D modeling. In this subsection, the author provides a brief description of the rendering methodologies to get a clearer picture of the potential they can have by coupling them to a functional–structural model.

2.3.1 Particle Systems and Procedural Methods

The particle systems for the generation of trees were created by Reeves [128] to represent in synthetic images the remarkable variety of irregular shapes and random variations exhibited by natural objects. Instead of using classical representations based on surfaces such as polygons, patches and quadratic surfaces, particle systems represent objects as a cloud of primitive particles that occupy the volume [128]. The algorithm consists of two successive steps. First, the general characteristics of the object are specified through a database; then, the generation of details comes in to describe the object completely. Details such as branches, buds, twigs, and leaves are generated in this phase. In the particle systems that shape trees, each tree is traced as a set of segments and small circles representing the branches and leaves, respectively. A set of initial features and dimensions, such as the height of the tree, is stochastically assigned before invoking the recursive branch generation procedure. The values of these parameters are randomly determined by distributions associated with the type of tree, according to phyllotaxis. Particle systems can model solid objects that are conventionally modeled with surface-based techniques. An example of the use of particle system technique to generate vegetation and trees is the *SpeedTree*® commercial software [129]. This software is made specifically for video game and movie development, and the graphic rendering is photorealistic. The focus is more

on the visual results than on the biological modeling, although the generated trees appear very realistic as shown in figure 2.7.

FIGURE 2.7: Tree generated with *SpeedTree*® software using particle system

Weber and Penn [62] proposed in 1995 a new approach to tree rendering based on particle systems with the goal of creating synthetic trees as realistically as possible. Weber and Penn's procedure needs a set of approximately 50 input parameters such as stem splits, stem children, stem radius, leaves, pruning, vertical attraction, leaf orientation, and degradation at a certain range. The approximation of the model refers to simplification of tree shape and structure in order to minimize the number of parameters. The general appearance and the lower part size of the tree are the main parameters needed to start with tree generation. Then, branching levels are considered as well as trunk foot shapes. The algorithm continues for every maximal tree branching level using an average value in a range variation (i.e., child branch angle, deviation angle, length relative to father branch, number of branches, photropism) [130]. One of the most popular softwares using this approximate approach is *Arbaro* [131]. An example of a generated tree with *Arbaro* in shown in figure 2.8.

FIGURE 2.8: Eastern cottonwood tree generated with *Arbaro* software using Weber and Penn's approximate approach

2.3.2 Rule-Based Methods

Rule-based methods differ from procedural algorithms by implementing a formal rule in order to transform a primary state into a final one with the application of a number of changes. This supplies a compact description of complex final conditions, even if generating geometric data from a set of rules is an intensive computational task. Only recent computational advancements made it possible to work with the vast amount of data associated with large objects in near real time. Lindenmayer systems (L-systems) for the generation of tree geometry, described thoroughly in [14], are widely used as rule-based method as described in the previous section (*L-PEACH* [55], *V-Mango* [28], *MAppleT* [51]). In addition to the models already described, L-systems are also used in commercial 3D modeling software such as *OnyxTree* from Onyx Computing [132]. Trees generated by this software are used only as visual elements in 3D art productions: they are visually very similar to trees (Figure 2.9), but they are lacking from the point of view of adherence to biological rules and interaction with the virtual environment, since this is not their purpose. However, they express all the potential that these technologies can put at the service of the FSPM models already analyzed.

FIGURE 2.9: Tree generated with OnyxTree® software using L-systems generation method

Another rule-based technique is the combination of the latter and procedural techniques, generating the shape as a combination of unitary components such as leaves, trunk and branch parts, and fruits. The distributions are produced using multiplication components that have built-in parameters defining the number of objects to be generated, their distribution characteristics, and their orientation. By connecting the components, the plant is defined as a directional graph. The graph represents the system of rules, and its edges describe the production dependencies. After the geometry of a parent component is generated, the production of the geometry for all the children is invoked [16] until the entire descriptive graph is processed. In this way, the components can be interlinked freely, and recursions are also possible. One software that uses this generation method is *XFrog* from XFrog Inc. [133] (Figure 2.10). Nevertheless, as in other software, *XFrog* lacks a biological rule base. The positive thing about rule-based object production methods is that most plants on earth can be generated with a small number of components or parameters providing a user-friendly interface, being a good base for a future link between 3D rendering systems and FSPMs.

FIGURE 2.10: Red Oak tree generated with *XFrog* software

2.3.3 Hybrid Methods

Hybrid 3D modeling methods are those that attempt a link between biological growth rules and 3D shape generation rules. Some FSPM models discussed in this review can be considered as hybrids in a broad sense as they combine a functional–structural model with a 3D modeling part of the results obtained. From a 3D modeling software point of view, there are some noteworthy ones that try to create this link between biological rules (minimal and often not scientifically correct) and 3D modeling, showing the potential of the method. Laubwerk software developed *Plants Kit 14*, a 3D rendering plug-in for commercial software that claims to be based on botanical/biological rules [134]. In *Plants Kit 14*, the operator has the control only on 3D rendering parameters such as level of details (LOD), polygon density, and switching between "summer, fall, winter, spring" tree presets. Despite the claim of being based on botanical/biological rules, the operator has no control on these rules (Figure 2.11).

FIGURE 2.11: Acero negundo rendered with *Plants Kit 14* by Laubwerk software

The extensive research by Pirk et al. and the University of Kostanz [135, 20] on 3D modeling of vegetation has led to the creation of a method to model synthetic trees to be interactive with surroundings (e.g., obstacles) or extract plausible growing steps. The generated trees can be used in 3D rendering software such as Maya, 3DS Max, and Houdini but are not for scientific applications. *TheGrove3D* is a commercial plug-in for Blender (Blender Foundation) [136] open-source 3D rendering software. It can be considered as based on hybrid method because of the use of both procedural rendering techniques and biological models. Trees are generated according to some pre-set parameters (e.g., internode growth length, branch angle), with some of them that are species-specific. The biological model is based on a simulation of tree growth mechanisms such as branch elongation. The model considers the number of nodes and growing buds regulate their own growth by controlling the flow of sugars and hormones. The plug-in can grow and modify trees according to spatial occlusion or regulate growth in presence or absence of light and shadow (Figure 2.12).

FIGURE 2.12: Ponderosa Pine Hill generated with *TheGrove3D* plug-in for Blender

Hybrid models are the proof that an interface between the world of photorealistic 3D rendering and mathematical biological models is possible. The potentialities, as discussed, are numerous for both fields. For the modelers, it is possible to visualize the result of their models in a satisfying way, besides having a more precise feedback from the virtual environment, thanks to a structure very adherent to the reality, and besides the not less important calculation of some environmental parameters, such as the quantity of light, which is possible to calculate in a more precise way with a 3D model. For the 3D artists, it is possible on the other hand to have even more realistic trees because they are strictly adherent to biological rules of growth.

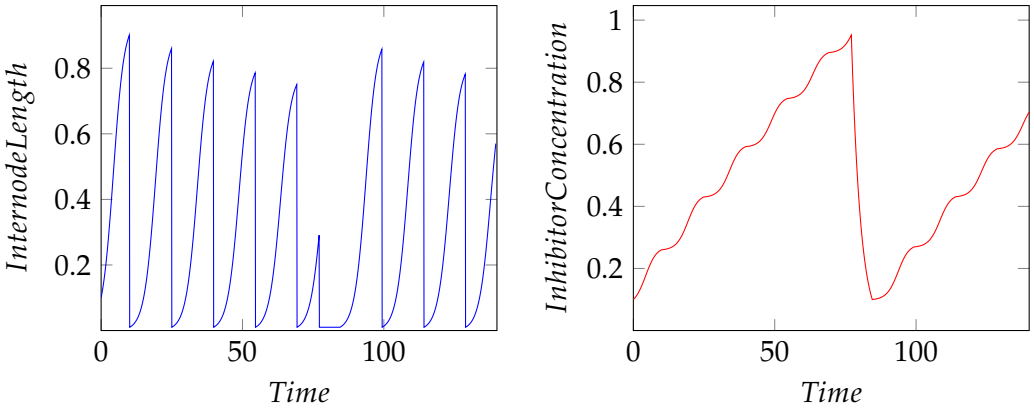
2.4 Scientific Proposal and First Proof of Concept

In this section, a scientific proposal is introducted for a link between a biological mathematical model based on Ordinary Differential Equations (ODE) and a 3D real-time rendering engine and the results of first experiments with the model, showing how a synthetic tree reacts to different light amounts in a 3D real-time environment, first with a simple simulation of an internode growing in different light conditions (different growing rate), then expanding the model to the whole tree, showing how it grows differently under different virtual environmental light conditions. This link can operate in both ways, i.e., by rendering a tree using parameters from ODE solutions or by considering geometrical 3D data (e.g., the amount of light from the virtual scene) as input to solve the ODEs. We also provide the starting mathematical biological model in *Simile* [137] and how we linked this model to the real-time 3D rendering engine. The proposed model is presented in more detail, with its development, detailed results and potential agronomic applications in chapter 3.

In particular, the final tree is made up of several independent and interconnected modules, following object-oriented programming (OOP) paradigms:

- A biological mathematical model to simulate the internode and branch growth;
- A 3D structure module to render the tree.

The mathematical model is a set of ordinary differential equations able to simulate the internode elongation and the rules for stopping growth. The internode elongation is a function of environmental parameters (temperature and light) and some species parameters (growth rate and max length). Stopping rules of internode growth are simulated according to the concentration of an inhibitory substance produced during the internode elongation. Internodes grow until a given inhibitor concentration is reached; then, the tree enters a seasonal growth stop and starts growing again after the inhibitor concentration reaches a minimum (Figure 2.13). Restarting the growth, the system will generate one or more branches depending on the species-specific parameters set. Figure 2.14 shows the first proposal of stock and flow model representation in the system dynamics tool *Simile* [137].



(A) Internode length in function of simulation time

(B) Inhibitor concentration in function of simulation time

FIGURE 2.13: Internode length and inhibitor concentration first simulations. (**A**) Internode length; (**B**) inhibitor concentration.

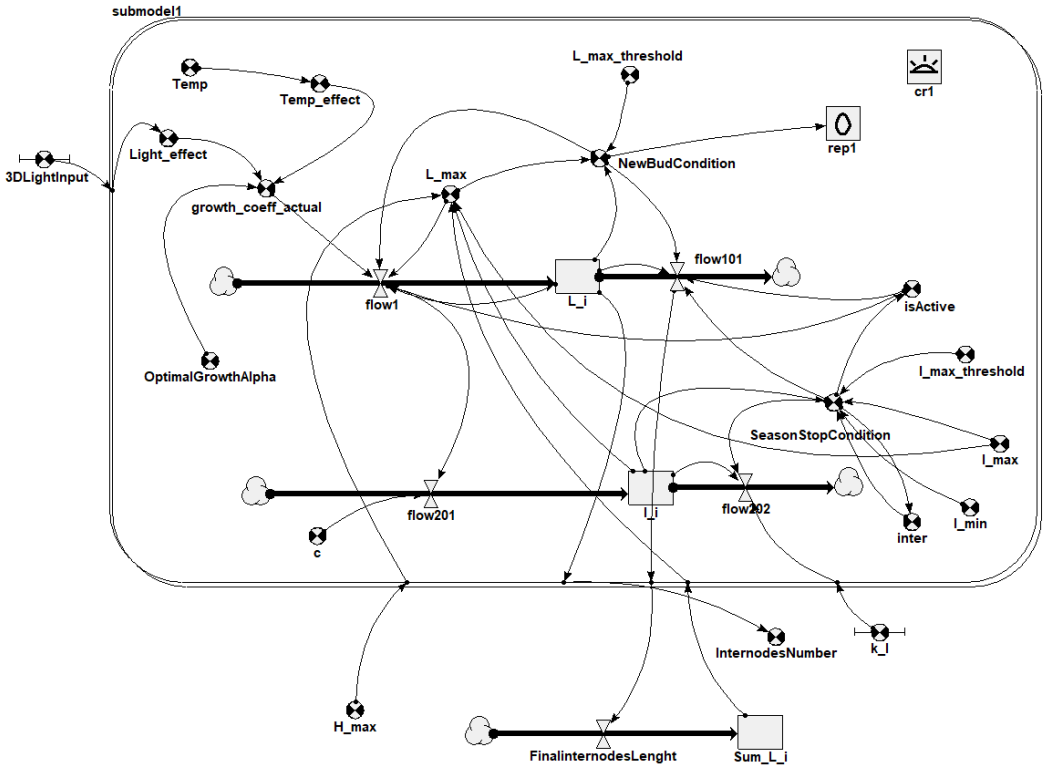
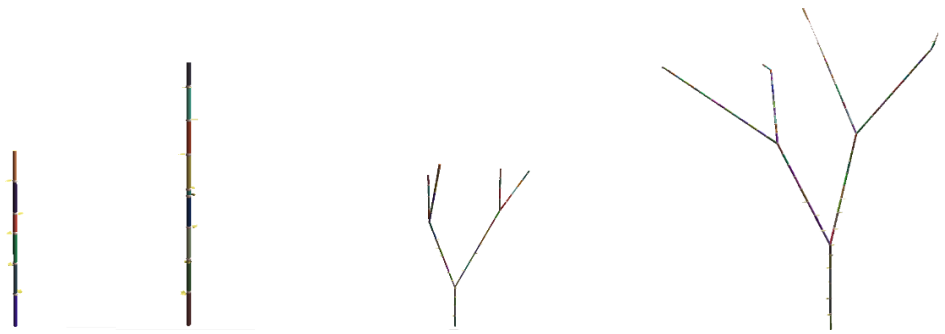


FIGURE 2.14: First proposal of stock and flow model of internode growth in the system dynamics tool *Simile*.

The 3D module is composed by two different sub-modules: (**a**) internode module, finalized with the creation of the internode geometry (length and width) thanks to the data calculated by the integrator, which contains the stop-growth rules (maximum length reached) and the positioning of the apical bud; (**b**) bud module, modeled as a particular internode of unitary length, which stores information about the number of children to be generated, the branching angle, and the state of the bud (asleep/awake). A particular bud module called seed stores species-specific information about the tree and is hierarchically the first bud from which all subsequent buds, branches, and internodes descend. The modeling mathematical part is independent from the rendering part due to its modularity: the change of mathematical model does not affect the 3D generation part. The simulation is capable of calculating the amount of light in the virtual environment to be used as a parameter in the mathematical model of growth. Global illumination, radiosity, and ray-tracing solutions have been tested, each with their own precision and computational performance, leading to the conclusion to use a custom shader for the calculation of the amount of light. This solution is a balance between accuracy and computational performance, sufficient for the purpose of simulation. An ambient light sensor has been programmed that renders a temporary texture on the surface of the sensor itself, extracting an array of RGB values and converting them into brightness values. The calculated value of the amount of light is used as a parameter in the calculation of the length of the internode in the relative module. A higher light amount leads

to higher internode growth, and vice versa lower light amount leads to lower internode growth, as shown in the first results (Figure 2.15). More detailed results and development of the model are provided in chapter 3. The results of the first tests with the model are shown in Figure 2.15. A first simulation was performed with a simple one-child internode model at two different light conditions. In Figure 2.15a, it can be seen that internodes grow differently depending on the amount of ambient light in which they grow. The left internode has a shorter total length than the right internode as it grew in a low light environment. The result will be different with different amounts of light in terms of single internode length and total length. In the presence of branches, as shown in Figure 2.15b,c, even a simple tree behaves differently. Specifically, it was set as number of children equal to 2 and branching angle of 30° (the model allows to set species-specific parameters depending on the tree to be modeled, with the number of children and branching angle being some of them). From Figure 2.15b,c, it can be seen that the amount of light affects the total length of the tree and the lengths of individual internodes. Branching is triggered when the calculation of the inhibitor concentration, referred to the individual branch, reaches its maximum value. Once the maximum concentration is reached, the tree enters a seasonal growth stop phase until the inhibitor concentration reaches a minimum value. Upon reaching the minimum value, the tree resumes growth by generating the number of new branches set oriented according to the branching angle (Figure 2.13b). In addition, the length of the single internode decreases as the total height of the tree increases, in accordance with the model. Internodes at the top of the tree will be shorter than those on the main trunk (the first to grow from seed) because they are closer to the maximum height of the tree.



(A) (B) (C)

FIGURE 2.15: Different growing rate and ramification of internodes under different light conditions: (**a**) Amount of annual growth under different light conditions, with low ambient light (in shadow) on the left, and full ambient light on the right; (**b**) ramification process in low light conditions, with different colors used to show different internodes; (**c**) ramification process in full light conditions, with colors used to show different internodes.

2.5 Conclusions

As seen in previous sections, FSPM models can be used as a basis for creating 3D models of trees that are consistent with biological rules. However, not all models analyzed have 3D structure output that can be satisfactory from a visual or functional point of view. Techniques for 3D modeling of photorealistic trees do exist, as analyzed in section 2.3, and we believe that an integration between the two worlds may be possible that can enhance both the field of mathematical biological modeling by being able to calculate some parameters more precisely, and the field of 3D modeling that can make use of a strong scientific basis to generate even more likely trees. A proof of concept of this integration has been presented in this paper, showing how it is possible to create a synthetic tree in a 3D real-time modeling environment from a mathematical biological model. The flexibility of the model is given by the use of interconnected modules according to the object-oriented programming (OOP) paradigm. The use of a proprietary shader for calculating the amount of light in the virtual environment provides further flexibility to the model. It is possible, in fact, to use the calculated amount of light to modify the main growth parameters such as branching angles and amount of growth. In this way, it is also possible to model the competition for light between near trees in addition to the change in branching direction caused by a tree's own shadow. The proposed model is presented in more detail, with its development, detailed results and potential agronomic applications in chapter 3.

2.6 Scientific work produced

[138] - Mariano Crimaldi, Edoardo Pasolli, and Francesco Giannino. "Use of process-based system dynamic models to generate biologically aware 3D trees". In: 9th International Conference on Functional-Structural Plant Models: FSPM2020.Hannover: Institute of Horticultural Production Systems, Oct. 2020, pp. 117–118

[139] - Mariano Crimaldi, Fabrizio Cartenì, Francesco Giannino. VISmaF: Synthetic tree for immersive virtual visualization in smart farming, in Proceedings of the 1st International Electronic Conference on Agronomy, 3–17 May 2021, MDPI: Basel, Switzerland, DOI: 10.3390/IECAG2021-09880

[140] - Mariano Crimaldi, Fabrizio Cartenì, and Francesco Giannino. "VISmaF: Synthetic Tree for Immersive Virtual Visualization in Smart Farming. Part I: Scientific Background Review and Model Proposal". In: Agronomy 11.12 (Dec.2021). Number: 12 Publisher: Multidisciplinary Digital Publishing Institute, p. 2458. DOI: 10.3390/agronomy11122458.

Chapter 3

Real-Time 3D tree model development, results and potential agronomic applications

3.1 Introduction

Biological-mathematical tree models can be used for a wide range of agronomic applications including crop management, visualization of ecosystem changes over time, field phenotyping, crop loading effects, plant function testing, biomechanics, and many others [141, 142, 50]. Some models propose a 3D tree output that, in addition to having functionality to visualize the result, offers an additional tool for the evaluation of some parameters of the model itself (light interception and quantity, temperature, obstacles, physical competition between multiple trees)[143]. Among the various biological-mathematical models, those called Functional-Structural Plant Models (FSPMs) in the last two decades have been developed by scientists to explore and integrate plant structure and its biological/functional processes [4]. The complexity of 3D rendering has reached high levels of quality standards [16] that allow the use of 3D structure not only as a simple visual output of FSPMs but also to characterize plant phenotypes [17] or as feedback to evaluate and calculate light partitioning [18]. 3D plants can be part of several modeling systems, being major components of many digital representations of landscapes or natural sceneries [19]. A 3D tree can be used in several scientific applications, such as synthetic forestry [21], generic digital representation of the real world [22], flow dynamics [23] and, as mentioned, in botany to determine light distribution and physiological parameters [6, 24]. Geometric modeling of plants allows researchers to visually validate biological processes, such as how plants interact with light and the environment. Among the biological-mathematical models that provide 3D output, a biological-mathematical model of a tree with a 3D output of its structure in a real-time 3D rendering (*Unity*©) environment is presented in this paper after proposing a proof of concept in Part I [140]. In section 3.2, the biological-mathematical model used is described (section 3.2.2), how it was implemented in code according to the object-oriented programming paradigm and in *Unity*© (section 3.2.3), how a custom shader was developed

to calculate the amount of light, and how virtual environmental parameters were used as model input (section 3.2.6). The results obtained are presented in section 3.3 with possible combinations and variations between species-specific tree parameters, virtual environment parameters, and competition between neighboring trees. In section 3.4 the results and possible agronomic applications of the model presented in this work are discussed (3.4.1).

3.2 Materials and Methods

3.2.1 Model Description

The proposed model is based on a hybrid approach with IBM (Individual-Based Model) elements and ODE elements interacting with each other [144]. The plant is simulated as a set of internodes whose internal dynamics of each internode are described by ODE, while the production of new internodes and interactions with the external environment are managed by the IBM part. This model (section 3.2.2) is linked to *Unity*© [145] rendering engine in order to provide a real-time, procedural 3D output of model calculations. The link can operate in both ways, i.e., by rendering a tree using parameters from ODE solutions, or by considering geometrical 3D data (e.g. the amount of light from the virtual scene) as input to solve the ODEs. *Unity*© has been chosen as 3D rendering engine because it offers a programming flexibility given by the use of scripts in C# language and a series of tools for the management of geometries and their interactions (collisions, physics, hierarchy, etc.). *Unity*© is also a widely used environment for many applications, mainly for the production and development of videogames. Since it was designed as a video game development environment, it has been necessary to program from scratch some features necessary for the correct functionality of the model, such as a custom shader for the calculation of the virtual environmental light but more importantly an algorithm for the numerical resolution of the ODEs involved. In the following subsections is shown how the model has been developed starting from the stock and flow model representation in the System Dynamics tool *Simile* [137] and then all transferred in *Unity*©, developing the modules in C# adhering to the Object-Oriented Programming (OOP) paradigm (3.2.3). Being a real-time 3D rendering engine, through the development in *Unity*© it has been possible to implement the variation of the virtual environmental parameters during the run-time: it is not necessary to stop the simulation in order to change these parameters and then to begin again but it is possible to vary them in real-time observing instantly how the model adapts, changing the input parameters and computing the modified tree architecture.

3.2.2 Growth model

The mathematical model representing the growth dynamics of the virtual plant's internodes has been developed as a set of ordinary differential equations (ODEs)

representing the dynamics in time of three state variables: length (L) and width (D) of the internode, and the accumulation of growth-associated inhibitory compounds (I) which regulate the activity/dormancy states of the meristems within the organ. More specifically, inhibitory compounds concentration (I) is considered representing the production of metabolic by-products accumulating within tissues that exert a negative feedback on meristematic cellular activity. Thus, seasonally this accumulation increases until growth is stopped (seasonal stop) and then resumes at the end of the stop when all waste products are cleared.

TABLE 3.1: Variables involved in tree growth ODEs system.

Variable	Definition	Assigned value
L	Length of internode	$0.1(t=0)$
I	Concentration of inhibitor	$0.1(t=0)$
D	Width of internode (secondary growth)	$0.1(t=0)$
t	Simulation time	0
c_L	Optimal growth coefficient	0.5
c_I	Optimal inhibitor coefficient	0.2
c_D	Optimal secondary growth coefficient	0.01
$envTemp$	Virtual environmental temperature	as set by user
$inputLight$	Light amount	-
H_{max}	Max tree height	species-specific
I_{max}	Max inhibitor concentration	1
I_{min}	Minimum inhibitor concentration	0.1
D_{max}	Max internode width	5
k_i	Season stop parameter	0.1

Specifically, the differential equation modeling the *i-th* internode length over time is defined as follows:

$$\frac{dL_i}{dt} = \alpha \ * \ L_i \ * \ \left(1 - \frac{L_i}{L_{max}}\right) \tag{3.1}$$

where:

$$\alpha = c_L * envTemp * inputLight \tag{3.2}$$

with c_L representing the optimal growth coefficient; $envTemp$ as temperature given by virtual environment and $inputLight$ as the amount of light calculated into virtual environment. L_{max} represents the maximum length of the internode calculated as:

$$L_{max} = \ 1 - \frac{\sum L_i}{H_{max}} - 0.2 * \frac{I}{I_{max}} \tag{3.3}$$

where $\sum L_i$ is the actual tree height, H_{max} is the tree max height as species-specific parameter, I is the inhibitor concentration as calculated in equation 3.4, I_{max} is the set maximum value of inhibitor concentration. The differential equation modeling

inhibitor concentration is defined as follows:

$$\frac{dI}{dt} = c_I * \alpha * L_i * \left(1 - \frac{L_i}{L_{max}}\right) - k_i * I \tag{3.4}$$

where c_I is the optimal inhibitor coefficient, k_i is a given parameter that changes if the tree is in seasonal stop or not. The parameter k_i changes its value if the inhibitor concentration reaches the I_{max} value or the I_{min} value, making the tree enter and exit season stop, as follows:

$$\begin{cases} I \geq 0.95 * I_{max} & \rightarrow k_i = 1 \quad \rightarrow season\ stop \\ I \leq I_{min} & \rightarrow k_i = 0 \quad \rightarrow restart\ growth \end{cases} \tag{3.5}$$

The differential equation modeling secondary growth is defined as follows:

$$\frac{dD_i}{dt} = d_{actual} * D_i * \left(1 - \frac{D_i}{D_{max}}\right) * k_i \tag{3.6}$$

where d_{actual} is a parameter function of virtual environment temperature calculated as:

$$d_{actual} = c_D * envTemp * inputLight \tag{3.7}$$

where c_D is the optimal secondary growth coefficient. D_{max} is the maximum value set for secondary growth and k_i is a given parameter as in equation 3.5. An overview of all variables involved, their definitions and assigned values are shown in table 3.1. This set of differential equations cannot be solved directly in *Unity©* because there is no built-in differential equation solver. A C# script as abstract class has been written that allows the above differential equations to be solved via the fourth-order Runge-Kutta numerical method. This script has been associated with the internode element for integration of its length and secondary growth, and with the branch element for calculation of the inhibitor concentration. For example, the internode element processes its length at each time step by numerically solving the associated differential equation (equation 3.1) at each time step thanks to the integration script created.

The internode elongation is then simulated as a function of environmental parameters (temperature and light) and some species-specific parameters (e.g., growth rate and max length). Stopping rules of internode growth are simulated according to the concentration of an inhibitory substance produced during the internode elongation. Internodes grow until a given inhibitor concentration is reached, then the tree enters a seasonal growth stop and then starts growing again after the inhibitor concentration reaches a minimum. Restarting the growth, the system will generate one or more branches depending on the species-specific parameters set (number of children).

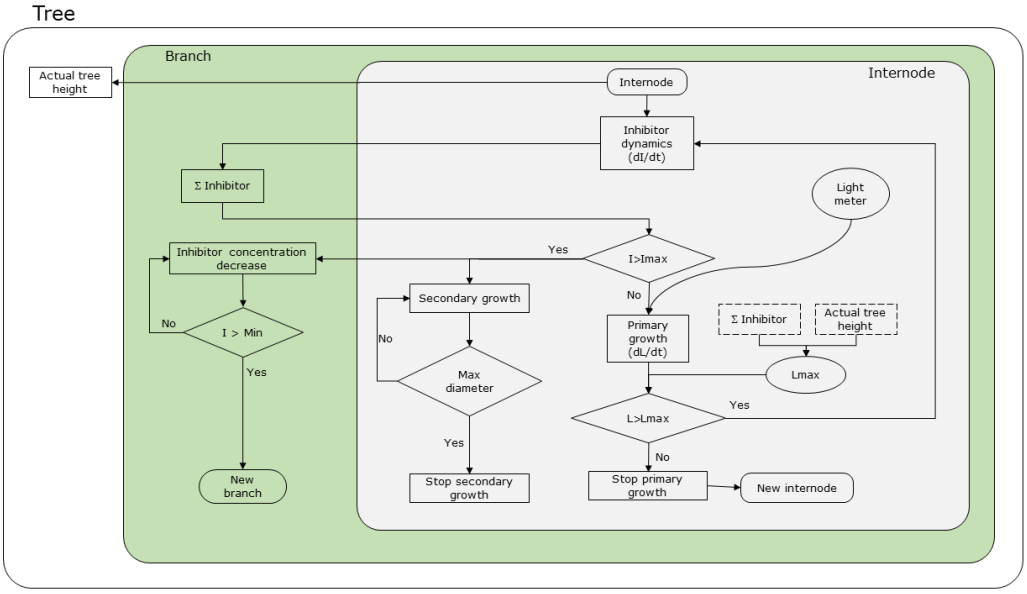


FIGURE 3.1: Proposed tree growth model scheme and relationships between modules.

3.2.3 Code structure and Unity3d development

The model shown in figure 3.1, a concise explanation of the model developed in *Simile* (presented in the previous chapter 2) that translates in stock and flow representation a system of ordinary differential equations that resolve numerically time variable dimensions, describes as flow-chart the separation of working modules and their mutual interactions. As discussed in the work by Crimaldi et al. [140], the choice of a real-time 3D engine was decided on *Unity*© as it has some useful features for the purpose. *Unity*© uses the C# language to code scripts associated with elements (prefabs) in the working environment. Each element (or module) has a linked C# script that defines its behavior. The C# language complies with the object-oriented programming (OOP) paradigm: each element can inherit all common features, methods, functions and parameters from another. *Unity*© also allows the management of the hierarchy between elements enabling to have elements as father generating children, a situation that is well suited to modeling a tree where there will be child internodes descending from common father. Moreover, thanks to the use of *Unity*© as a real-time rendering engine, it is possible to change some parameters during run-time, parameters such as the virtual environment temperature; quantity, direction and light characteristics allowing the modeling of some typical situations of a tree growth like the cycle of day and night or the seasonal cycle.

As shown in figure 3.1 the tree has a number of parameters that allow the model to run. Some are species specific, others are process parameters that change step by step in real time as the simulation runs. The species-specific parameters are set at the beginning of the simulation and do not change during the simulation. They are

parameters proper of the tree species that is intended to simulate. In the model proposed in this work it has been chosen to simulate as species-specific characteristics the branching angle (defined as the angle of the new branch created in relation to the parent), the number of children (defined as the number of new branches created after the end of the seasonal stop of the tree, figure 3.1), the maximum height of the tree (defined as the maximum height to which the tree will tend asymptotically, figure 3.1) and the simulation time (defined as how long the simulation will run, 1 second in simulation equals to 1 day of tree life). These parameters were coded in the C# script shown in Listing 1 as dumb code of *Tree* class.

```
public class Tree : MonoBehaviour {
    // Number of children
    public int NumChildren = 2;

    // Ramification angle
    public float RamAngle = 30f;

    // Max tree height
    public int maxTreeHeight = 100;

    // Lists of Branches, Internodes and Nodes (buds)
    public List<TreeNode> nodes = new List<TreeNode>();
    public List<Internode> internodes = new List<Internode>();
    public List<Bud> buds = new List<Bud>();

void Start() {
     // Start tree with species-specific parameters;
    }

void Update() {
     // Update tree each time step until simulation time has reached;
    }
}
```

LISTING 1: Tree script with species-specific parameters settings

These settings are possible in *Unity*© modifying the variables of the script via the inspector (a built-in *Unity*© editor), figure 3.2, or via scripting. According to the object-oriented programming paradigm, the other modules and sub-modules will descend from the *tree* script, inheriting the common characteristics. Each module that inherits from *tree* will have the same characteristics, for example it will inherit the maximum height of the tree as a parameter. The *tree* module holds also lists of all branches, internodes and buds (nodes) of tree to process the tree hierarchy.

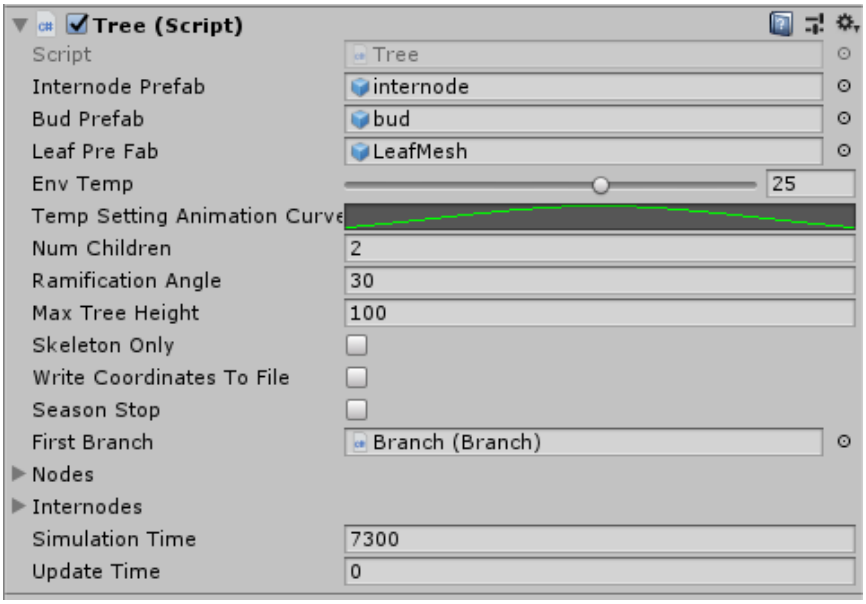


FIGURE 3.2: The species-specific parameters (number of children, ramification angle, max tree height) modifiable via *Unity*© inspector panel. The "simulation time" parameter sets the temporal length of the simulation (1 second = 1 day)

The tree has a final structure formed by a series of nodes (lateral and apical buds) and consequential internodes according to a hierarchy (figure 3.3). The main trunk is the collection of main internodes father of all subsequent child internodes that have a lower hierarchical scale up to the apical internodes that are the internodes positioned lower in the hierarchical scale. Branches are the entities that contain the entire sequence of nodes and internodes and that manage, thanks to their code scripts, the functions of generating new branches at certain branching angles and the concentration of inhibitor that will determine the seasonal stop of the tree. At the end of the seasonal stop, as shown in figure 3.1, new branches are generated according to the number of children set in the species-specific characteristics of the tree and according to the branching angle. The actual branching angle deviates from the ideal branching angle set as a species-specific parameter based on the amount of light calculated from the virtual environment as explained in subsection 3.2.6.

The *Internode* C# class manages the behavior of internode explained above. As shown in the dumb code in Listing 2, the internode can have 3 states: awake, asleep or dead. The three states can be changed thanks to external events or by other modules of the tree (i.e., by branch module at the end of the seasonal stop). The internode only grows when it is in the awake state, is paused when it is in sleep state (it can resume growth), and can no longer grow if it is in the dead state. At tree initiation, the initial numerical conditions of the differential equations for calculating length and width (secondary growth) are set as well as the creation of a *GameObject* (a fundamental object in Unity) "end" representing the end point of the internode whose position is the numerical solution of the ODE that calculates the length (see method `public override void Init(Tree t)` in Listing 2). This approach has been chosen

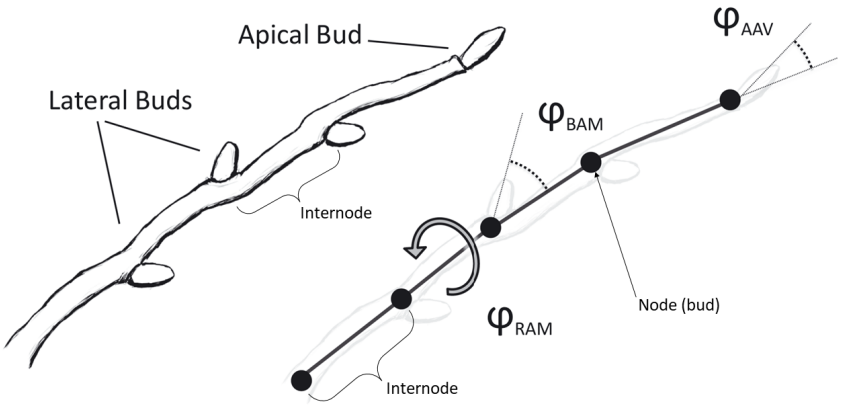


FIGURE 3.3: Nodes-internodes scheme of a branch with branching angles. φ_{RAM} is the bud rotation angle around internode axis; φ_{BAM} is the bud angle to internode axis; φ_{AAV} is the apical bud angle to internode axis

because in this way it is not created a 3D shape representing an internode and then scaled according to the calculated length, but only the position of the end point is calculated, reducing the computational load according to the principle of procedural rendering as proposed by Weber and Penn [62].

The length of internode, e.g. the position of "end" *GameObject*, is calculated in method `public void UpdateGrowth()` as well as the max internode length (as in equation 3.3). The *Internode* class takes also as input parameter the light amount (subsection 3.2.6) and temperature form virtual environment.
The `private IEnumerator UpdateSecGrowth()` is a peculiar class called *coroutine*. The *coroutine* is used in *Unity*© as a method that can pause execution and return control to *Unity*© but then continue where it left off on the following update. This means that the action that takes place within a *coroutine* does not happen within a single update like in standard methods. The use of *coroutine* is well suited to the computation of secondary growth that must be updated during the season stop of the tree and one step at a time.

The *Branch* C# class manages the behavior of branch(es) as explained before (equation 3.4). As shown in the dumb code in Listing 3, the branch can have 3 states: awake, asleep or dead like the internode. The branch calculates the inhibitor concentration numerically resolving the ODE (equation 3.4) setting the numeric initial conditions with method `public void BranchInit(Tree t, Bud b)`, then calculates the numeric solution of the ODE at each time step performing the method `private void InhibitorUpdate()`. This method stops when the inhibitor concentration reaches its maximum and puts the tree (and the branch) in seasonal stop. When the inhibitor concentration reaches its minimum, the method ends the seasonal stop of the tree and restarts inhibitor concentration calculations, as well as re-start growing the tree. While in seasonal stop, a *coroutine* method is called

```
public class Internode : Tree{
    // Internode state
    public enum InternodeState {
        Asleep, Awake, Dead
    }

    private IEnumerator UpdateSecGrowth() {
        // Secondary growth ODE solution calculation via Integrator script
    }

    public override void Init(Tree t) {
        // Create "end" object
        // Set internode length initial conditions for ODE integration
        // Set secondary growth initial conditions for ODE integration
    }

    public void InitInternode(Tree t, Branch b) {
        // Create Light Meter instance at internode creation
    }

    public override void InitGrow() {
        // Start internode growth
    }

    public Bud InitBud() {
        // Start and put bud in place
    }

    public void UpdateGrowth() {
        // Calculate internode length at each same step as ODE solution
        // Calculate max internode length in function of actual length and max tree height
        // Take as input the light amount and temperature
        // Create a new bud at the end of growth and destroy light meter instance
    }
}
```

LISTING 2: Internode script with all methods performing length calculation and other variables and parameters

(`private IEnumerator ChangeSeason()`) that waits until the seasonal stop ends, then creates new branches as the number of children set.

The *Bud* C# class manages the behavior of buds, both apical and lateral ones (figure 3.3). As for internodes and branches, buds can have three states: awake, asleep and dead. In this case, as shown in Listing 4, thanks to the update method `public void Update()`, the script will check at each update if an external event affects the buds, changing their state from asleep to awake or dead and vice versa. If the bud is awakened by an external event, it will produce new branches calling the method `public Branch CreateBranch()`. The new branches will be rotated at certain angle function of light amount. The same method is called by *Branch* script when seasonal stop ends.

The method `public Internode CreateInternode()` is called when the internode stops growing having reached its maximum length. This method generates a new internode child of the previous one with a rotation angle function of the amount of light (φ_{AAV} in figure 3.3). Another important function that is handled by the *Bud* class is leaf placement. In this work we decided not to model the leaves as an organ of the plant with its own ODEs but only as an object in the 3D environment

```
public class Branch : Tree {
    public enum BranchState {
        Asleep, Awake, Dead
    }

    public void BranchInit(Tree t, Bud b) {
        // Set branch inhibitor initial conditions for ODE integration
        // Add branch in tree branches list
    }

    private IEnumerator ChangeSeason() {
        while (seasonStop)
        {
            // Stop while season stop
        }
        for (int i = 1; i < tree.NumChildren; i++)
        {
            // Create new branch(es) for every bud in the tree
        }

        // Ends season stop and restart growing
    }

    public override void BranchUpdate() {
        // Update branch(es) and inhibitor concentration
    }

    private void InhibitorUpdate() {
        // Calculate inhibitor concentration at each same step as ODE solution

        if (!seasonStop && (annualInib >= 0.95 * inhibitorIntegrator.IMax))
        {
            // Activate season stop when inhibitor concentration reaches max
        }

        if (seasonStop && (annualInib <= inhibitorIntegrator.IMin))
        {
            // Restart growing when inhibitor concentration reaches min
        }
    }
}
```

LISTING 3: Branch script with all methods performing inhibitor concentration calculation and other variables and parameters

with its own mesh, orientation and rotation. This is because, during preliminary tests, we found an exponential decay of computational performance as the number of leaves increases. The leaves, therefore, having their own shape and position, will contribute to the calculation of light, influencing tree own shadow and consequently the growth of the tree. In the *Bud* class the leaf position is managed by the `public override void Init(Tree t)` method, which manages the angle and position on the bud itself. These parameters are either fixed as hard coded value or depend on the position of the bud. A particular bud is the one named *seed* in the hierarchy: it is considered as the first bud from which the main trunk grows. The *Bud* class will call the `public override void InitGrow()` method if the bud is the first (seed) and generate the first branch as the main trunk.

In *Unity*© engine, the whole structure of the tree is based on a hierarchy between

```
public class Bud : Tree {
    public enum BudState {
        Asleep, Awake, Dead
    }

    public void Update() {
        if (externalConditionOccured && state == BudState.Awake)
        {
            // Bud state changes to awake if it was set as asleep
            state = BudState.Awake;
            // Create a new branch from awaken bud
        }
    }

    public Internode CreateInternode() {
        // Create a new internode from bud after the father internode has stopped growing
        var angle = // Function of light amount
        // The new internode created is rotated as the angle function of light amount
        return internode;
    }

    public Branch CreateBranch() {
        // Create the number of new branches set
        var angle = // Function of light amount
        // New branches created are rotated at certain angle of ramification function
        // of light amount
        return branch;
    }

    public override void Init(Tree t) {
        base.Init(t);
        float leafAngle = // Depends on bud position or hard coded

        // Leaf Placing
        if (!geometry && !tree.skeletonOnly)
        {
            // Place leaf prefab on bud at position
        }
    }

    public override void InitGrow() {
        if (Seed || tree.NumChildren == 1)
        {
            // Create first branch (main trunk) from first bud (seed)
        }
    }
}
```

LISTING 4: Bud script with all methods performing internode creation, branch creation, leaf placement and other variables and parameters

the elements described above. Thanks to C# scripts and the object-oriented programming paradigm, all elements are interconnected at the structure level itself but also as behavior. Any single element can affect all other elements and vice versa. The single elements in *Unity*© that have a shape and a behavior are called *GameObject*. The figure 3.4 shows a hierarchy of a tree in the very first phase of growth in order to visualize a simple hierarchy that will evolve in a much more complex form. It can be noticed how from the *GameObject Tree* all the other *GameObject*(s) descend hierarchically, starting from the first bud called *seed* and then moving on to subsequent branches, internodes and buds. As explained above, the *GameObject* "end"

is the one positioned in the space where it should be the end of the length of the growing internode calculated step by step by solving the ODE (equation 3.1). The *GameObject* "internode (clone)" is the one in charge of creating the mesh (or geometry) of the internode from the starting point (the bud that generates it) to the end point ("end"). By doing so, although it seems an artificial procedure, it is possible to separate and make independent the calculation of the length of the internode (e.g. the position of the *GameObject* "end") and the mesh or geometry of the same internode. The *LightMeter* in the hierarchy is the *GameObject* in charge of calculating the amount of light from the external virtual environment, it will be described in detail in the next paragraph.

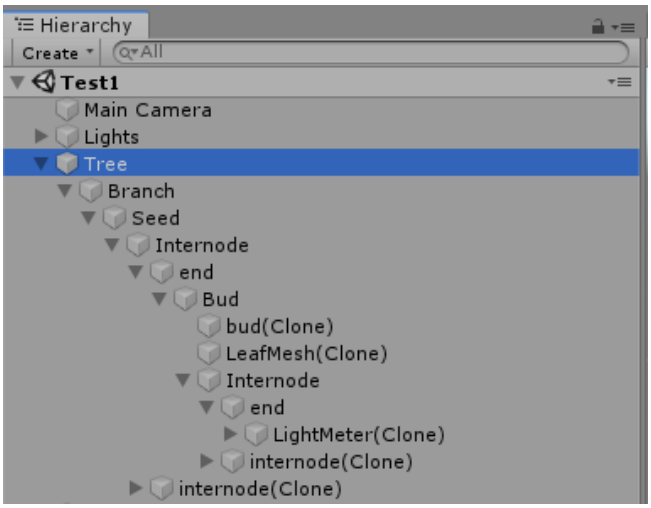


FIGURE 3.4: Hierarchy in *Unity*© showing a simple early tree structure: one branch, two internodes, two buds

3.2.4 Numerical ODE integration method

The differential equations governing internode growth, inhibitor concentration, and secondary growth (equations 3.1, 3.4, 3.2.2) must be solved numerically because they do not have an explicit solution. There are several numerical methods for solving the differential equations, each with its own approximations, boundary conditions, and numerical errors. Of the many, the fourth-order Runge-Kutta method [146] was chosen to be used in this work. In *Unity*© there is no module dedicated to the resolution of differential equations, so it was necessary to program a script specialized in the numerical resolution of ODEs using the numerical method of fourth-order Runge-Kutta. As shown in dumb code in listing 5 at each time step, the integrator function is passed the current values of internode length, inhibitor concentration and secondary growth so that the next time step is calculated having as initial value (initial condition) the value at the previous time step. Having a code specialized only in the integration of ODEs and separated from the rest allows a better allocation of computational resources as it is possible to parallelize the process especially when

the number of internodes and branches is very high and it is necessary to calculate thousands of numerical solutions simultaneously.

```
abstract public class Integrator //: MonoBehaviour
{
    // ODEs settings: x are values being integrated, xdot derivatives being calculated
    abstract public void RatesOfChange(double[] x, double[] xdot, double t);

    /// Step forward using 4th order Runge Kutta method
    // x are values being integrated
    // h is the time step
    public double RK4Step(double[] x, double t, double h)
    {
        RatesOfChange(x, k1, t);
        for (int i = 0; i < nEquations; i++)
        {
            store[i] = x[i] + k1[i] * h / 2.0;
        }
        RatesOfChange(store, k2, t);
        for (int i = 0; i < nEquations; i++)
        {
            store[i] = x[i] + k2[i] * h / 2.0;
        }
        RatesOfChange(store, k3, t);
        for (int i = 0; i < nEquations; i++)
        {
            store[i] = x[i] + k3[i] * h;
        }
        RatesOfChange(store, k4, t);
        for (int i = 0; i < nEquations; i++)
        {
            x[i] = x[i] + (k1[i] + 2.0 * k2[i] + 2.0 * k3[i] + k4[i]) * h / 6.0;
        }
        return t + h;
    }
}
```

LISTING 5: Integrator script with Runge-Kutta method to solve ODEs

3.2.5 Leaf Procedural Mesh Generator

As explained in the previous subsections, the leaves have not been modeled as an organ, therefore without a module that models their biological function, but are still present in the structure as simple meshes, positioned according to the rules managed by the bud module (see listing 4) and participating in the calculation of shadows as will be explained in the next paragraph (3.2.6). Since they are very numerous, especially in the final stages of growth, it was necessary to code them as procedural meshes, using a method that at each event of leaf positioning generates a mesh with a given shape and a predefined texture in the calculated position. With this solution, it is possible to optimize the computational load of the operation because in memory will not be stored the 3D shape of the leaf with its texture but only the "law" that generates its mesh, reducing the operations. In *Unity*© the procedural generation of a mesh follows the principle that each mesh has a defined number of nodes, triangles and the order in which these elements are counted. Triangles are considered by

counting vertices counterclockwise. In the specific case, having to create a rectangular mesh, the vertices to be generated will be 4, with 2 triangles. The dumb code in listing 6 shows how these vertices are created and how the triangles are defined. Once the mesh is generated, a texture will be applied to its surface, which will also handle transparencies through the alpha channel. The textured mesh will be placed at the point calculated by the bud module, as shown in the listing code.

```
public class LeafMeshGenerator : MonoBehaviour
{
    private void Start()
    {
        CreateShape();
        UpdateMesh();
    }

    void CreateShape()
    {
        vertices = new Vector3[]
        {
            // Creates an array of vertices in defined position
        };

        uv = new Vector2[]
        {
            // Creates an array of normals (positive faces of mesh)
        };

        triangles = new int[]
        {
            // Creates an array of triangles
        };
    }

    void UpdateMesh()
    {
        // Generates mesh from vertices, triangles and normals arrays
    }
}
```

LISTING 6: Procedural leaf mesh generator

3.2.6 Light amount calculation

A key part of the model development is to calculate the amount of light coming from the virtual environment. Several shader solutions have been tested to find the optimal one in terms of computational performance and flexibility. A shader is simply a program that runs in the graphics pipeline and tells the computer how to render each pixel. These programs are called shaders because they are often used to control lighting and shading effects [147]. Standard lighting shaders are unable to compute indirect light (such as diffusive light). The first shader tested was the one called Global Illumination (GI) which allows the calculation of diffuse light [147]. GI considers not only the light that comes directly from a given light source, but also how that light on impact with a material continues to bounce around the given scene. While very computationally efficient, the GI shader needs baked shadow to work.

The baked shadows are shadows pre-calculated at the beginning of the simulation and are not variable in real-time which makes the GI shader computationally performant. With the shader GI it is not possible to calculate the own shadow, which made us to discard it as a solution. Another shader tested was the so called *Radiosity*. This shader calculates the light between actual scene triangles, rather than between additional artificial structures (light probes, voxels etc). It is derived from heat transfer calculations. The surfaces of the scene to be rendered are each divided up into one or more smaller surfaces (patches) like finite-elements methods. A coefficient describing how well the patches can see each other is calculated. Patches that are far away from each other, or oriented at oblique angles relative to one another, will have smaller coefficients. Brightness is function of coefficient for each patch (figure 3.5) [147]. The *Radiosity* shader can calculate the indirect diffuse light even in real-time and dynamic scenes, is less performant than GI (depends on number of patches) but still relatively fast (using parallel GPU calculation) but the transition areas need a lot of patches, making the algorithm less performant. Also, some transition areas are hard to approximate with patches. Given the nature of a tree structure, with lots of light transition zones,this solution has been discarded.

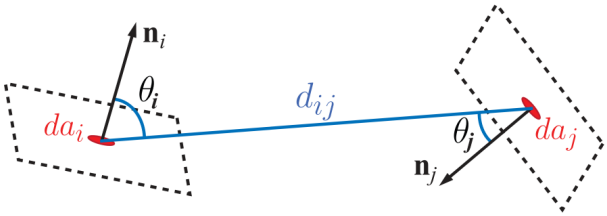


FIGURE 3.5: Parameters of two patches in *Radiosity* shader. d_{ij} is the distance between the patches d_{ai} and d_{aj}; n_i and n_j are the vectors normal of patches surface; θ_i and θ_j are the angles between d_{ij} and the vectors normal to the surface n_i and n_j

The *Ray-tracing* shader calculates each ray as physical entity. Each ray can bounce on surface so, according to physical properties of the object hit, a new set of multiple rays is calculated. Having the physical properties of rays and object materials, it is possible to calculate the amount of light precisely, even indirect, diffuse, irradiated, reflected and diffracted light. The shader is very performance heavy and requires dedicated hardware and shader optimization [147]. Despite being the most accurate shader of all, given the nature of the tree structure, we noticed a decay in computational performance that prompted us to look for an alternative solution. The alternative solution, evaluating all the solutions previously shown, has been to write a custom shader specialized in calculating the amount of light. A "light meter" sphere *GameObject* is placed in the scene with a camera pointed on the object. The position of the camera changes each frame to cover the entire surface. The camera renders a very narrow texture and records into an array the values of RGB. Then, a conversion from RGB values to brightness is made thanks to equation 3.8 from ITU-R BT.709

standard [148] as shown in Listing 7 as dumb code. The final value of brightness will be the sum of all arrays calculated at each camera positions.

$$Brightness = 0.2126 * R + 0.7152 * G + 0.0722 * B \tag{3.8}$$

```
public class LightMeter : MonoBehaviour
{
    void Start()
    {
        // Define camera positions: front, back, left, right, up, down
    }

    void Update()
    {
        // Take pixel array values and calculate brightness
    }

    private void GetPix() {
        // Render temporany texture at i-th position of camera

        // RGB to brightness conversion of temporany texture
    }

    public void SumPix() {
        Output = // Final brightness as sum of all i-th temporany textures brightness values
    }
}
```

LISTING 7: Light meter script to calulare brightness from RGB array of values

This way it is possible to calculate direct and indirect light, even if the object is partially in shadow, in real-time and with a little weight on performance (if the number of cameras is kept low).

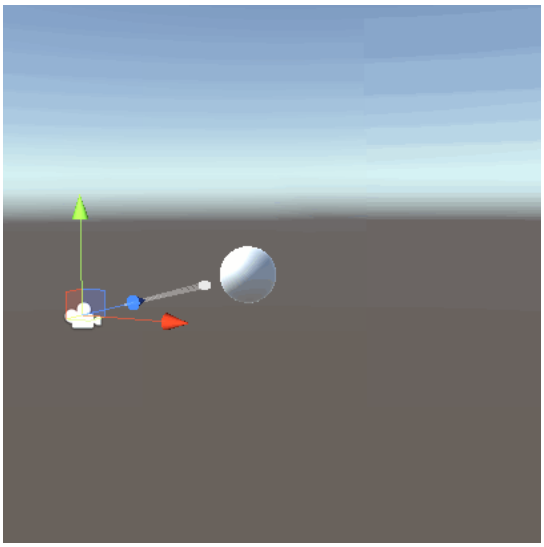


FIGURE 3.6: Light Meter *GameObject* with texture rendering camera in one position

As shown in equations 3.1 and 3.2 as well as in the growing scheme in figure 3.1, the light input affects internode growth with a standardized calculated value between 0 and 1. When the internode is growing in full light, with a value close to 1, its growth will be maximum or very close to maximum. The light meter was placed at the top of the growing internode, and then removed whenever the internode stops growing, as it is no longer necessary to calculate the amount of light. By doing so, it was possible to keep a relatively low number of light meters simultaneously active during growth, managing to have a certain balance of computational performance. In addition, by placing the light meter at the apex of the growing internode, it is possible to vary the growth angle φ_{AAV} shown in the figure 3.3, which will deviate from the ideal one as a function of the amount of light calculated.

3.2.7 Virtual environmental model inputs

The *Unity©* environment allows for parameters as model inputs that can be defined as environmental. In the previous paragraph it has been shown how it is possible to calculate the amount of light that is a part of the environmental parameters. *Unity©* allows the programming and modeling of other parameters that can be classified as environmental. As shown in the model diagram in figure 3.1, another environmental parameter that was chosen as an input to the model, in addition to the amount of light, is temperature. Obviously in a virtual environment it is not possible to record a real temperature but it is possible to model its trend and use the simulated numerical data as model input. In the case considered in this work, the temperature data is an input that influences the coefficient α in the growth of the internode (equations 3.1 and 3.2). In combination with the amount of light, therefore, the temperature will affect the amount of internode growth. In *Unity©*, the *Animation Curve* function was used to simulate the non-constant temperature trend (temperature changes during seasonal and daily cycles). As the name suggests, the *Animation Curve* is natively used as a function to manage the behavior of an animation. However, it is possible to use it for any parameter, value, variable in the model. As shown in figure 3.7, the script *Tree* parameter called *Temp Setting Animation Curve* is managed by a curve that varies the value along a time range. In the example case in question, the temperature varies over a period of time from 1 to 365 (simulating a year) starting from the minimum value of 15 reaching the peak at 23 and then decreasing again. Once the 365^{th} step has been passed, the cycle starts again from 1 as if the year was starting again. The temperature value will be different for each time-step which as mentioned corresponds to a day for each second of simulation.

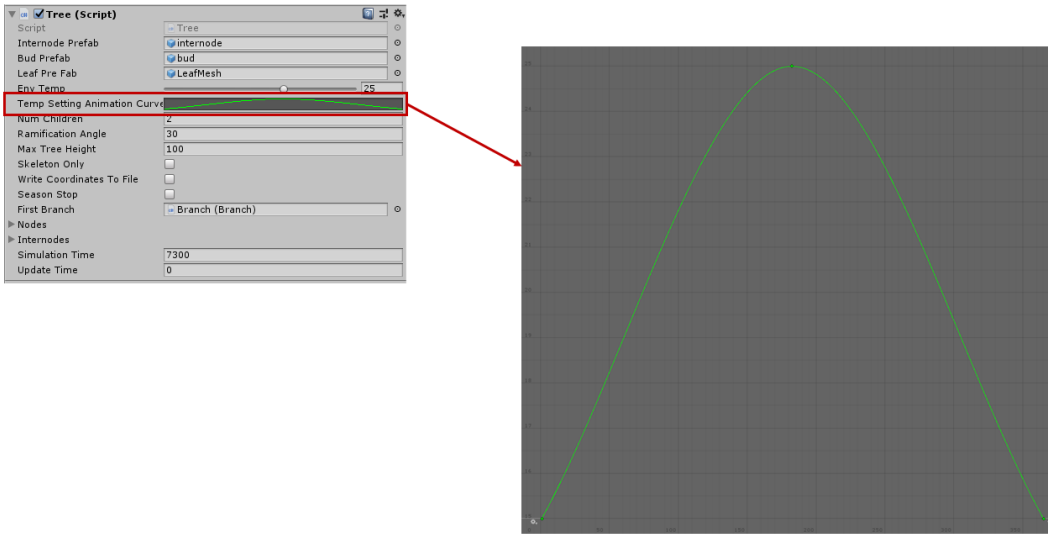


FIGURE 3.7: Temperature animation curve used to simulate the variability of temperature during the year

It is possible to create more complex curves to better simulate the real variability of temperature over time. Moreover, the animation curve can also be used to manage the variability of other environmental parameters such as the already mentioned ambient light in order to simulate the day/night cycle.

3.2.8 Experimental design

In order to test the potential of the model and its flexibility and adaptability to different conditions and parameter settings, experiments were conducted consisting of growing trees in the virtual environment created in *Unity*© with different combinations of both species-specific and environmental parameters. Specifically, simulations were first carried out leaving the environmental parameters fixed: with a constant temperature (set to 25°C) and amount of light (set to 1, corresponding to a tree growing in full light), only changing the species-specific parameters such as number of buds and branching angle. Specifically, tests were carried out with:

- number of buds: 2 - 3 - 4;
- branching angle: 30° - 40° - 55° - 60°.

Tests with a number of buds equal to 1 have already been done in the preliminary tests contained in the previous companion publication [140]. Tests made by changing environmental parameters were done by keeping a fixed number of buds and varying the temperature, the quantity and the direction of origin of the light source. The temperature regimes adopted in these simulations are as follows:

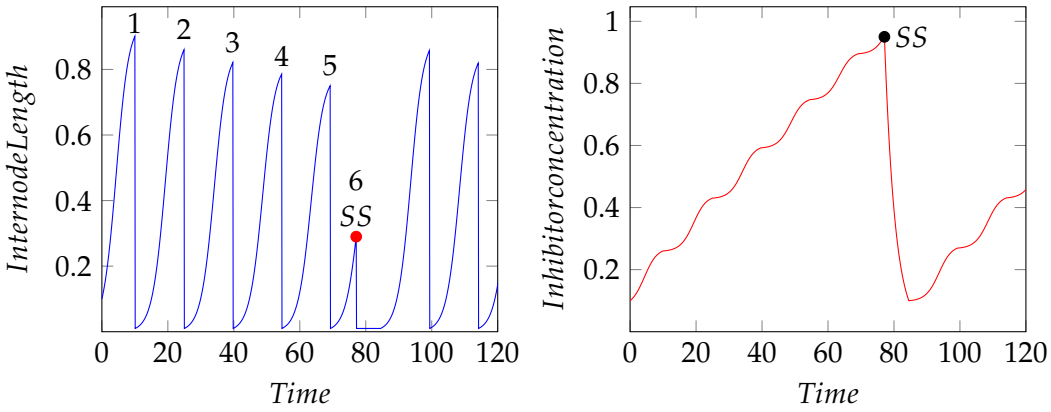
- fixed temperature at three different values: 15°C - 25°C - 35°C;
- variable temperature over time as explained in the previous paragraph (3.2.7).

Competition tests were carried out by growing 2, 3 and 4 trees with the same species-specific parameters and temperature at a variable distance (1m, 2m). Tree competition is for light as no module was modeled that simulates resources and thus competition for resources (e.g., nutrients, water). Finally, tests have been performed by varying species-specific parameters to simulate the growth of different species of trees; in this work, columnar-shaped trees (such as a *Populus nigra*) and conical-shaped trees (such as a *Picea abies*) have been modeled. All tests were carried out for a growth duration of 5 years (1826.25 seconds of simulation). In order to understand the expected behavior of the simulations, *Simile* computations were performed. The graphs produced by *Simile* software (see subsection 3.2.2) show the expected behaviors from the simulation, then confirmed by the model in *Unity*©. In the following subsections, the results with the different combinations of parameters discussed in the previous paragraph are shown. While computing the position of the leaves and considering them in the calculation of tree self shadow but not modeled as organs, the results of the simulations in *Unity*© are shown without the presence of the leaves to analyze and evaluate the architecture of the resulting trees with their species-specific characteristics.

3.3 Results

3.3.1 Species-specific parameter changes - fixed environmental parameters

The expected behavior, with fixed environmental parameters, is the one shown by the *Simile* simulations in figure 3.8. The tree grows by increasing internode length, until it reaches the maximum internode length calculated as in equations 3.1 and 3.3. At the end of growth, the internode stops growing, the apical bud generates a new child internode that will grow to its maximum length. Simultaneously, the inhibitor concentration increases until it reaches its maximum. When the maximum concentration value is reached, according to equation 3.4, the tree enters the seasonal stop and secondary growth starts. Inhibitor concentration begins to decrease until it reaches its minimum, the tree starts growing again, secondary growth stops, inhibitor concentration begins to increase again, as does internode growth. At the end of the seasonal stop, new branches will also be generated, of the same number as set in the species-specific parameters of the tree. The expected behavior of the simulation is, therefore, to have internodes that decrease their maximum length as they approach the maximum tree height. The result will be that the first internodes (those in the main trunk) will be longer than the apical internodes.



(A) Internode Length (B) Inhibitor Concentration

FIGURE 3.8: Internode length and Inhibitor concentration simulations in *Simile* as expected behaviour with fixed environmental parameters. *SS* = Season Stop. Numbers are the count of internodes

The result obtained in *Unity*© with fixed environmental parameters, 55° branching angle and 2 children (branches) is shown in figure 3.9. The tree reached after 5 years (1826.25 seconds) of simulation the height of 2.6*m*. It can be noticed that the number of internodes in the main trunk is the same as in the simulations in *Simile* (figure 3.8). It is possible to see the set branching angle and how the apical internodes are shorter than those in the main trunk.

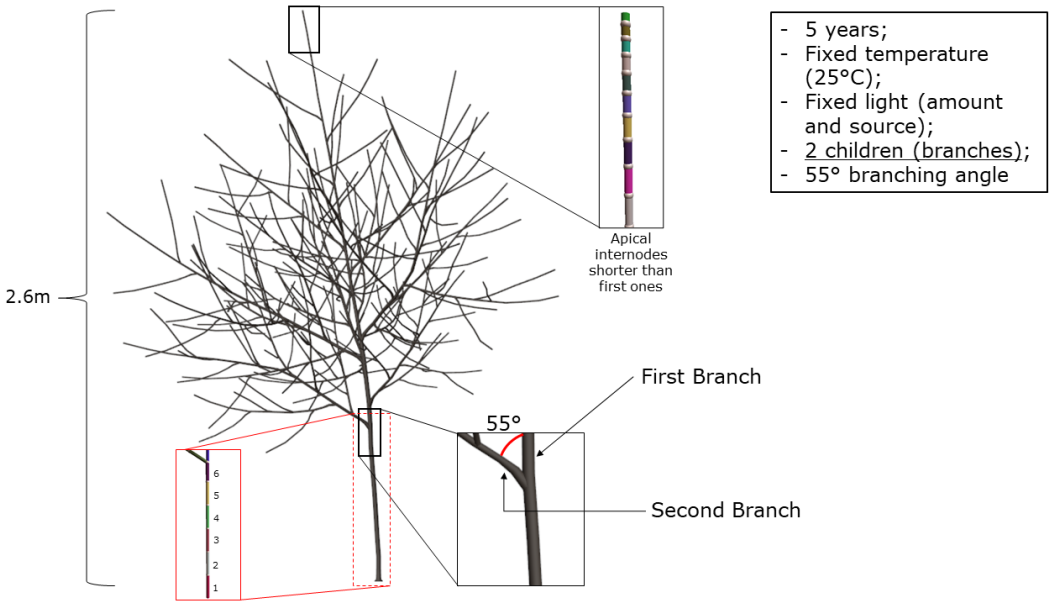


FIGURE 3.9: Tree growth with fixed temperature (25°C), fixed light source (top) and amount, 55° branching angle and 2 children branches. The apical internodes are displayed with different colors for showing purpose

By increasing the number of children, while leaving the environmental parameters constant, the tree appears more dense with branches as the number of new

branches generated is higher. The apical internodes remain shorter than those in the main trunk, as can be seen in figure 3.10 where the simulation generated a tree 2.55m in height.

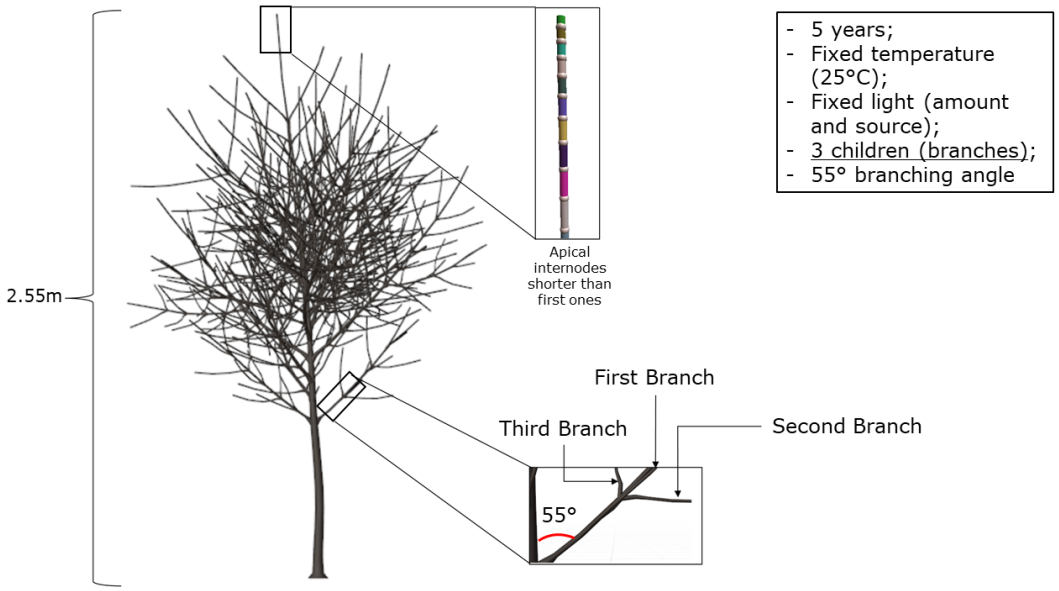


FIGURE 3.10: Tree growth with fixed temperature (25°C), fixed light source (top) and amount, 55° branching angle and 3 children branches. The apical internodes are displayed with different colors for showing purpose

Same effect generating a tree with 4 children, as shown in figure 3.11, where it was generated a tree 2.4m high.

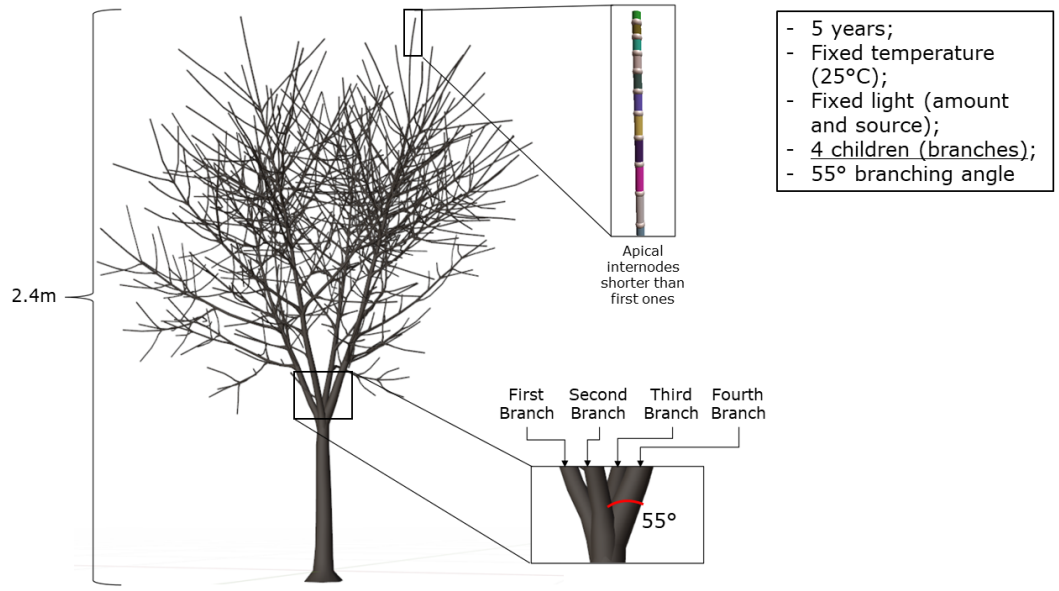


FIGURE 3.11: Tree growth with fixed temperature (25°C), fixed light source (top) and amount, 55° branching angle and 4 children branches. The apical internodes are displayed with different colors for showing purpose

By changing the branching angle instead, the results obtained are shown in figure 3.12. It can be seen that by changing the branching angle from 30° (figure 3.12a), 40° (figure 3.12b), and 60° (figure 3.12c) the tree widens the canopy and takes up more space. The heights are also different, respectively 2.4*m* for the trees with 30° and 60° branching angles, and 2.6*m* for the tree with 40° branching angle.

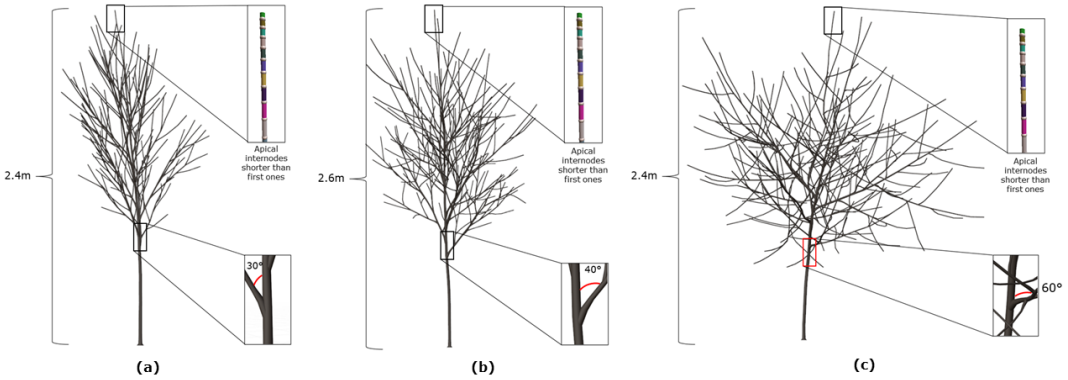
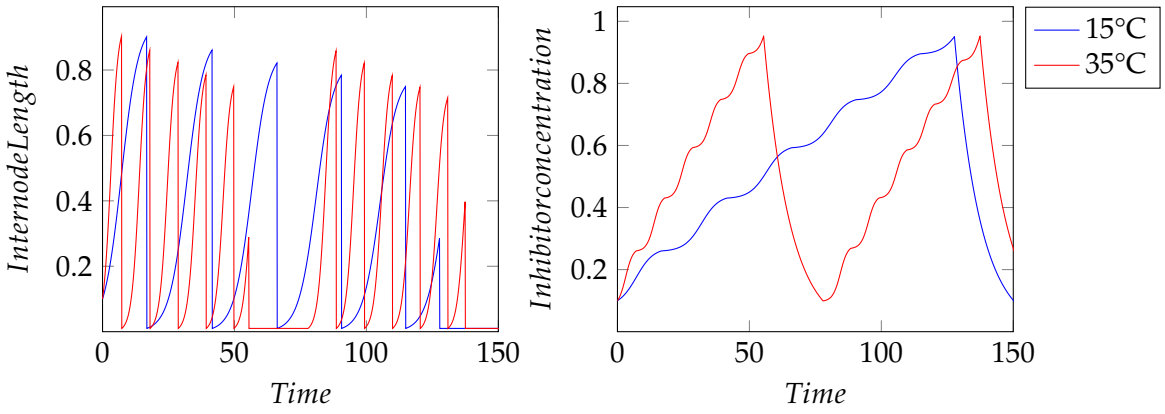


FIGURE 3.12: Tree growth with fixed temperature (25°C), fixed light source (top) and amount and 2 children branches. The branching angle showed is **(a)** 30°, **(b)** 40° and **(c)** 60°. The apical internodes are displayed with different colors for showing purpose

3.3.2 Virtual environment parameter changes

The expected behavior, changing the virtual environmental temperature, is the one shown by the *Simile* simulations in figure 3.13. By changing the temperature, the behaviors of the internode length and inhibitor concentration curves change. By lowering the temperature, internode growth is slower. The variation in inhibitor concentration also changes, slowing it down as well. At lower temperature, therefore, we will have the effect of seeing a tree grow, for the same simulation time, lower and less branched because it takes longer to reach the value of maximum internode length and maximum inhibitor concentration, which causes the tree to enter seasonal stop late, generating at lower heights the new branches. The result of growing a tree at a low temperature is shown in figure 3.14b where it also possible to see that the first branching is lower in height than the tree growing at higher temperature in figure 3.14a.

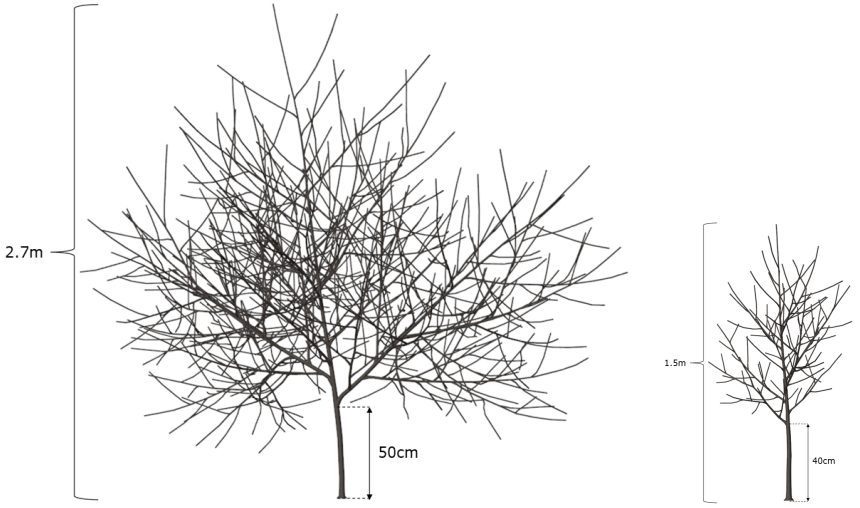


(A) Internode length with different environmental temperatures

(B) Inhibitor concentration with different environmental temperatures

FIGURE 3.13: Internode length and Inhibitor concentration simulations in *Simile* as expected behaviour with changing environmental temperature

By contrast, increasing temperature will cause internode length and inhibitor concentration to increase faster, reaching their maximum values sooner. We will have the effect of seeing a tree grow, for the same simulation time, higher and more branched because it reaches the value of maximum internode length and maximum inhibitor concentration quickly, which causes the tree to enter seasonal stop soon, generating at higher heights the new branches, as shown in figure 3.14a.



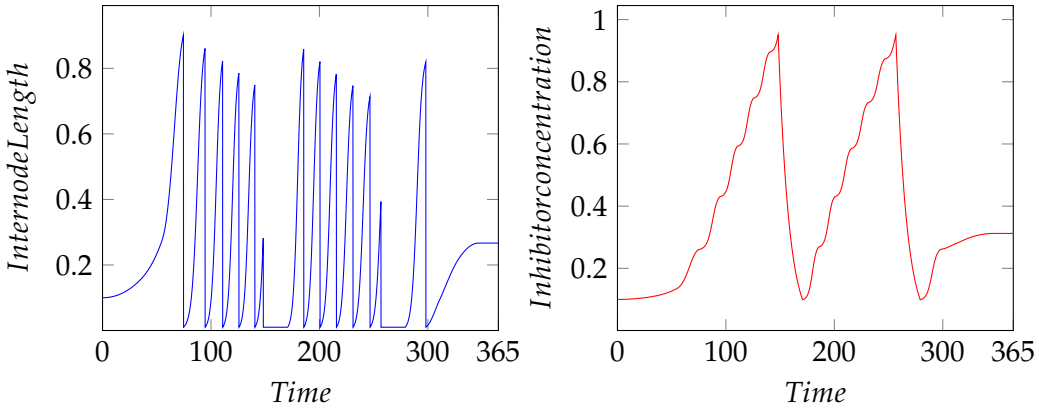
(A) 35°C environment temperature

(B) 15°C environment temperature

FIGURE 3.14: Tree growth with changing environmental temperature, fixed light source (top) and amount, 55° branching angle and 2 children branches

As described in subsection 3.2.7, the temperature can also be changed dynamically over time, following a variation curve. In this study, the variation curve is as

shown in figure 3.7: the temperature starts at second 1 (day 1) from a value of 15°C, reaches a maximum at second 182 (day 182) of 25°C and then drops back down to a value of 15°C at second 365 (day 365) and then starts again from day 1. The variation curve can have values and curvature at will, in this work it was set up this way to demonstrate how the growth function and thus the resulting tree adapts to changes in virtual environmental parameters in real time. From simulations in *Simile* showed in figure 3.15, it can be seen that the internode length curve slows down in the early part due to the lower starting temperature. As the temperature dynamically returns to the "standard" value of 25°C, the curve adapts and approaches the ideal (with fixed temperature) curve shown in Figure 9. After the second 182 (day 182) the temperature starts to decrease again and the curve changes in turn, slowing down the growth of the internode (figure 3.15a). The same trend is shown in the curve of the inhibitor concentration: slow increase in the first part, ideal situation in the middle part, slowed down in the second part where the temperature has a descending trend (figure 3.15b).



(A) Internode length with time variable environmental temperature

(B) Inhibitor concentration with time variable environmental temperature

FIGURE 3.15: Internode length and inhibitor concentration simulations in *Simile* as expected behaviour with time variable environmental temperature

The resulting tree with dynamically time changing temperature is shown in figure 3.16.

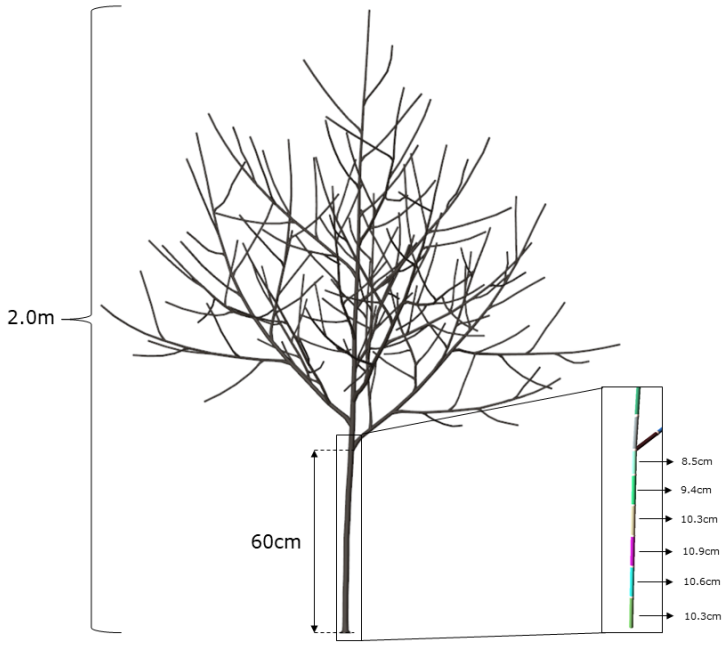
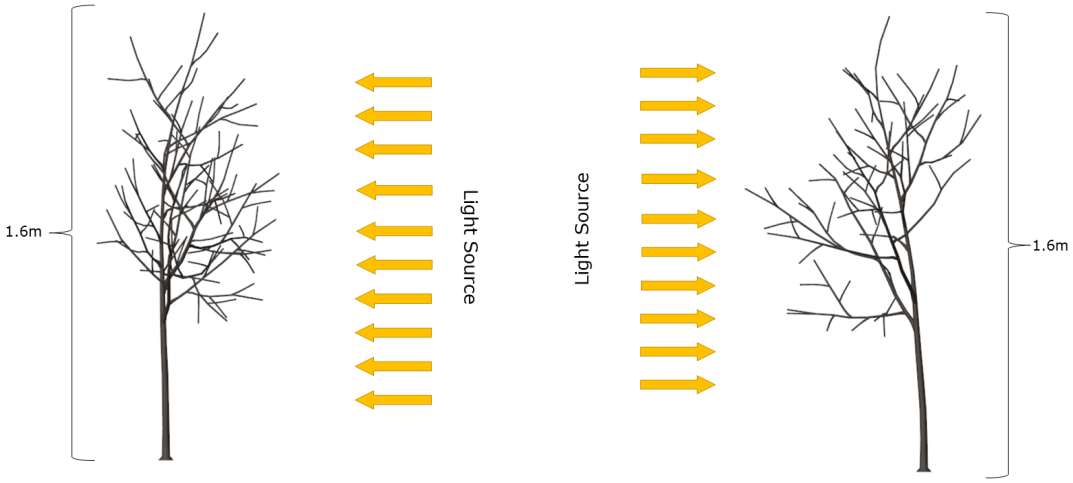


FIGURE 3.16: Tree growth with dynamically time changing environmental temperature, fixed light source (top) and amount, 55° branching angle and 2 children branches

It is possible to notice in figure 3.16 how the dynamic variation of the temperature makes the internodes grow in different measures according to the period in which the simulation is in the curve of variation of the temperature, in agreement with the model in *Simile* and the trends shown in figure 3.15. In addition, the tree is shorter in height ($2.0m$) than those grown under ideal conditions as growth is slowed at the periods when the temperature deviates from the ideal, increasing (first phase of the temperature change curve) and decreasing (second phase of the temperature change curve).

As shown in the synthetic model diagram in figure 3.1 and elaborated in the internode growth equations 3.1 and 3.2, the amount of virtual ambient light affects the rate of internode growth. The effect, as the amount of light changes, is quite similar to that shown in figure 3.13a, where as the temperature changes the rate of internode growth changes. Also in the case of the amount of light, if the amount of light is reduced, the growth rate of the internode will also be reduced, i.e. for the same simulation time the internode will grow more slowly and less. In addition to the amount of light, as explained in subsection 3.2.3, the direction of the light source will also affect growth. Specifically, internodes will grow by changing the angle φ_{AAV} (apical bud angle to internode axis, figure 3.3) to turn towards the light source. This effect can be clearly seen by placing a light source to the side, creating parts of the tree in full light and others in shadow, as shown in figure 3.17. The shadowed parts of the tree have internodes with a low growth rate due to low light, growing less, nothing at all if the light amount is zero. The brighter parts of the tree have higher internode growth rates as well as directing towards the light source. The

light not being equally distributed over the entire tree as in the previous results will also result in less overall tree growth, producing a smaller tree than ideal conditions.



(A) Tree growth with light source from right (B) Tree growth with light source from left

FIGURE 3.17: Tree growth with 55° branching angle, 2 children branches and different light sources

3.3.3 Competition between neighboring trees

The model's ability to adapt to the amount and location of the light source allows it to simulate even competing groups of nearby trees. Figure 3.18 shows the growth of 2 trees 1m away from each other while figure 3.19 shows 2 trees 2m away from each other. In figure 3.20 3 trees are shown growing 1m distance apart and in figure 3.21 2m distance apart. Figure 3.22 shows 4 trees grown in two parallel rows at 1m between trees and between rows. Lastly, figure 3.23 shows the growth of 4 trees arranged in 2 rows at a distance of 2m between trees and 2m between rows.

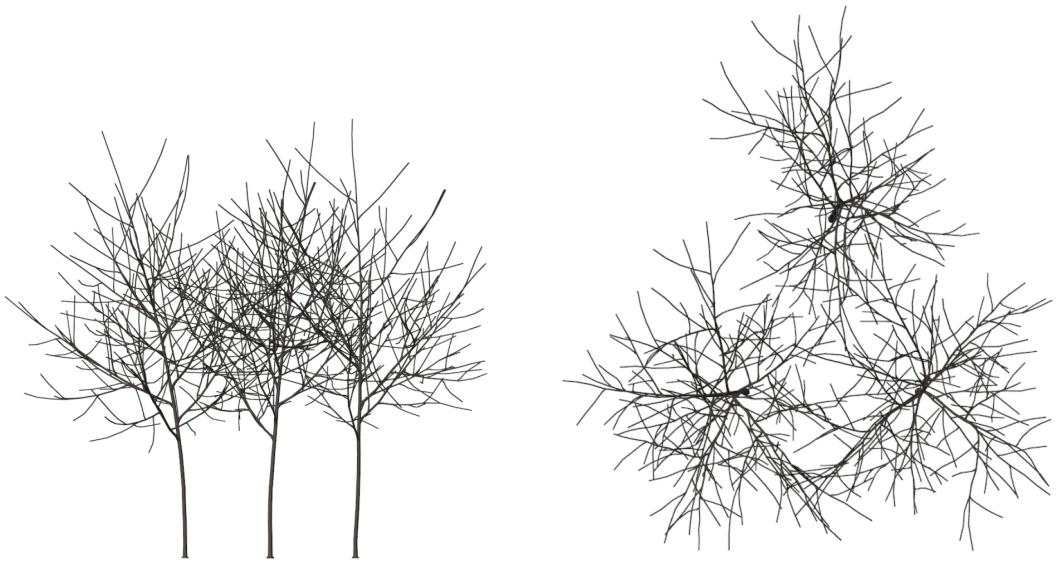
(A) Side view of 2 trees growing at 1m distance (B) Top view of 2 trees growing at 1m distance

FIGURE 3.18: Group of 2 trees growing at 1m distance

(A) Side view of 2 trees growing at 2m distance

(B) Top view of 2 trees growing at 2m distance

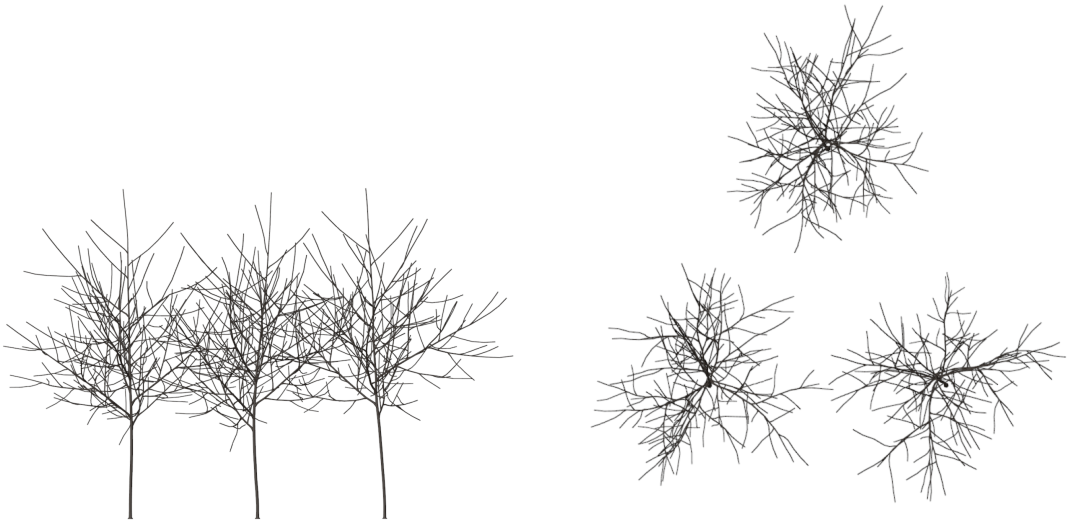
FIGURE 3.19: Group of 2 trees growing at 2m distance



(A) Side view of 3 trees growing at 1m distance

(B) Top view of 3 trees growing at 1m distance

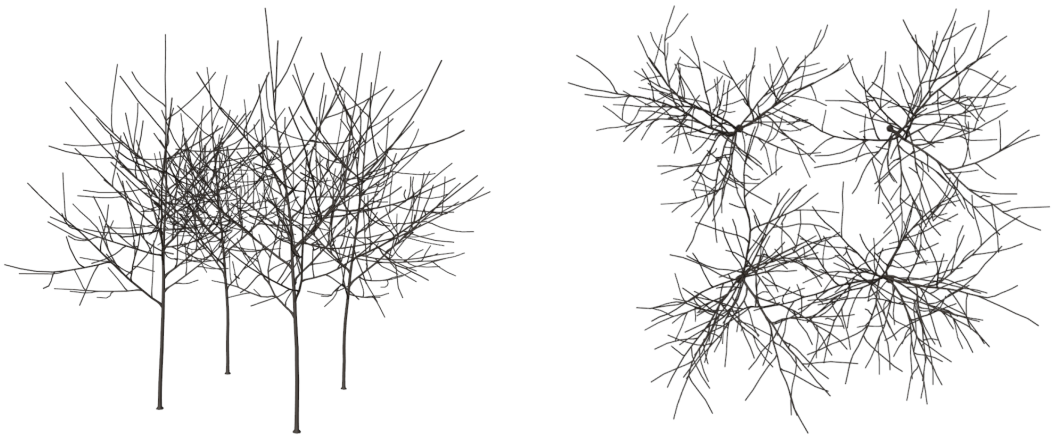
FIGURE 3.20: Group of 3 trees growing at 1m distance



(A) Side view of 3 trees growing at 2m distance

(B) Top view of 3 trees growing at 2m distance

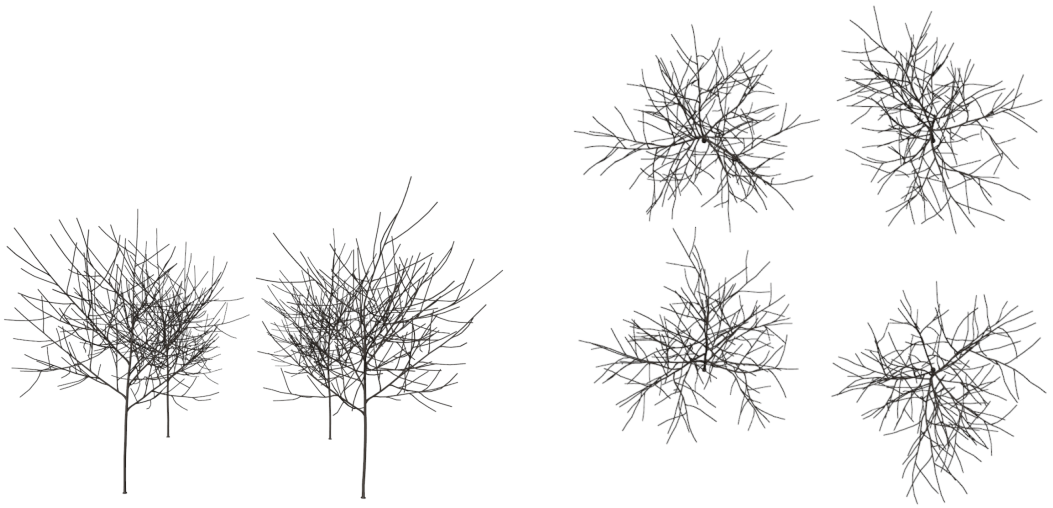
FIGURE 3.21: Group of 3 trees growing at 2m distance



(A) Side view of 4 trees growing at 1m distance

(B) Top view of 4 trees growing at 1m distance

FIGURE 3.22: Group of 4 trees growing at 1m distance



(A) Side view of 4 trees growing at 2m distance

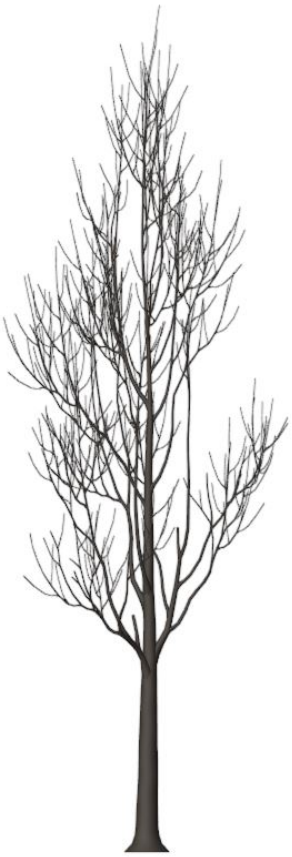
(B) Top view of 4 trees growing at 2m distance

FIGURE 3.23: Group of 4 trees growing at 2m distance

The greatest effects of competition for light can be seen in simulations with groups of trees at shorter distances from each other (1m) where the proximity of trees increases shade decreasing the rate of growth of internodes closer to the next tree and therefore more in shade. At higher distances the amount of shade generated by one tree on another is much less, causing neighboring trees to grow in near ideal conditions.

3.3.4 Model adaptation to different tree shapes

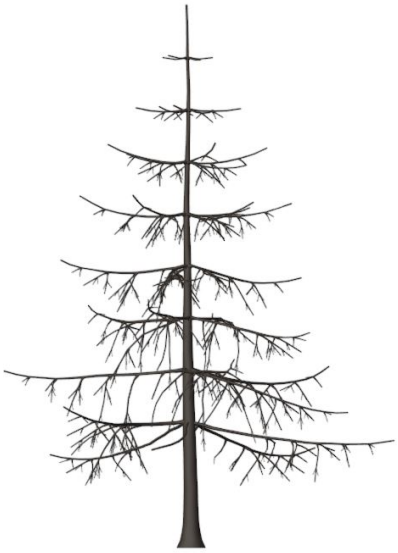
The tree models shown in the previous subsections all show cluttered globe-shaped trees. By changing the model parameters as described in the previous sections, you can model different types of tree shapes. In the examples proposed in figures 3.24 and 3.25, it is possible to see how by setting a branching angle of 45° and a number of children equal to 2, it was possible to recreate a tree with a columnar shape, as can be a tree of the species *Populus nigra* (figure 3.24a, with a comparison with a real photo of *Populus nigra* in figure 3.24b). Figure 3.25a shows a tree with a conical shape, with a number of children equal to 5 and a branching angle of 90° for the primary branches (those directly descended from the main trunk) and 55° for the secondary and tertiary branches (those descended from the primary or subsequent branches). In the case shown in figure 3.25a, it is possible to compare this type of tree with a real photo of a conifer with the same characteristics, such as a *Picea abies* in figure 3.25b.



(A) Columnar shaped tree with branching angle equal to 45° and number of children set equal to 2

(B) Real photo of a *Populus nigra* for comparison. Photo courtesy of T. Davis Sydnor, The Ohio State University, Bugwood.org

FIGURE 3.24: Visual comparison for columnar shaped tree



(A) Conical shaped tree with branching angle of 90° for the primary branches, 55° for the secondary and tertiary branches and number of children equal to 5

(B) Real photo of a *Picea abies* for comparison. Photo courtesy of Judy Slater, Bugwood.org

FIGURE 3.25: Visual comparison for conical shaped tree

3.4 Discussion

The results shown in section 3.3 show how the model is adaptable to different specifications, species-specific parameters, and environmental conditions. The flexibility given by the OOP paradigm allows a separation of modules (and their code) while remaining interconnected at the functional level. It is possible to change at any time any ODE that rules a specific module without having to modify the 3D rendering part nor affecting other parts of the code. Any change made can be immediately displayed in the generated output structure. In addition to ODEs, the results show that changes are also possible to the parameters that regulate growth, showing different outputs depending on the changes. By modifying species-specific parameters, it is possible to model potentially any tree species with different numbers of children, different maximum heights, and different branching angles. Being a real-time rendering environment and procedurally computing the solutions of the ODEs at each frame and rendering the results, it is also possible to modify the inputs of virtual environment parameters such as temperature and light. For the temperature, the

parameter is only numerical not being related to a physical component of the virtual environment and its variation modifies the amount of growth of the internode and therefore of the tree, as well as the concentration of inhibitor. As for light, in the virtual environment its value and position are linked to a physical component (light source). Changing the amount and position, therefore the direction of the light source, changes the rate of growth of the internode and therefore of the entire tree, also changing the direction of growth by simulating phototropism. Despite being a compromise between the precision of ray-tracing (but too heavy computationally) and the computational lightness of the GI (but not very accurate), the system adopted for the calculation of the amount of light turns out to be satisfactory for the aim of the work proving to be adaptable to the desired growth conditions. Although not shown in the results, the presence of the leaves contributes to the calculation of the self shadow and influences the amount of light measured. The leaves themselves were not modeled as an organ and do not have a specific module, something the authors will count on implementing in future work. Thanks to the calculation of the amount of light in real time, it was also possible to model small groups of trees competing with each other for light showing how the proximity and therefore the shadow generated by another tree, also affects the growth of the internodes (and the tree) modifying the amount of light detected. Competition for resources and their implementation is another module that the authors plan to incorporate in future work.

3.4.1 Potential agronomic applications of the model

The possible agronomic applications of the model presented in this work may be various. As shown in chapter 2 of other similar models, the potential of FSPMs with 3D output can be numerous. In the case of the model presented in this work, the authors count on being able to use it in applications to study the effects on the architecture of planting density, to calibrate allometric models, or even for quantitative analysis of branching patterns. Another potential use of the presented model, being in real-time, is to implement it in an immersive visualization system to train agricultural operators in the practice of pruning or more simply to evaluate the effects of pruning on tree structure where with the help of the model could be determined tree row volume (TRV), and based on that the appropriate application rate and dose. The authors are already working on a system that allows the "slicing" in real time of the meshes that are part of the tree structure to visualize the pruning operation and its effects. The modularity of the proposed system can also be used to implement effects of plant pathologies and visualize their consequences.

3.5 Conclusions

The use of FSPM models has been well established in the modeling community for two decades [24, 4]. Many models now have 3D output that allows rapid visualization of results as well as being an integral part of the model itself as a provider of input parameters, as described in chapter 2. The use of a real-time 3D rendering engine allowed the authors to create a link between a mathematical biological model of tree growth and the procedural creation of its architecture. This solution differs from other proposals of integration between models and 3D rendering in that is capable of having as input the parameters of the virtual environment in real time (amount of light and temperature), calculating at each variation (at each simulation time step) the solutions to the ODEs that regulate the processes showing immediately the result as a modification of the structure. While showing a flexibility that is well suited to the purpose, thanks to the implementation of interconnected modules adhering to the OOP paradigm specialized in individual components of the tree, the model has some margin for improvement such as the implementation of a module for the modeling of leaves as organs, or the integration of competition for resources in neighboring trees. Further studies are underway by the authors to implement the above features, as well as physical changes to the generated structure in the 3D environment such as a pruning operation.

3.6 Scientific work produced

[139] - Mariano Crimaldi, Fabrizio Cartenì, Francesco Giannino. VISmaF: Synthetic tree for immersive virtual visualization in smart farming, in Proceedings of the 1st International Electronic Conference on Agronomy, 3–17 May 2021, MDPI: Basel, Switzerland, DOI: 10.3390/IECAG2021-09880

[149] - Mariano Crimaldi, Fabrizio Cartenì, and Francesco Giannino. Integration of a System Dynamics model and 3D tree rendering - VISmaF Part II: Model Development, Results and Potential Agronomic Applications. SUBMITTED to Agronomy (2022)

Chapter 4

Deep Learning approach to sustainable weeds on-field management

4.1 Drone and sensor technology for sustainable weed management: a review

4.1.1 Introduction

Biotic threats such as insects, weeds, fungi, viruses, and bacteria can broadly affect crop yield and quality. Among these, weeds are the most impacting problem causing remarkable yield loss worldwide [150]. The most characterized effect of weeds is competition for resources such as light [151], water [152], space [153], and nutrients [154]. In addition, specific chemical signals and/or toxic molecules produced by weeds may interfere with a normal crop development [155]. A distinctive trait of wild species, including weeds, is their high physiological, morphological, and anatomical plasticity which makes them more tolerant than crop species to environmental stressors [156, 157, 158]. Moreover, weeds interact with other biological components of the environment, acting as refuge for plant pests such as insects, fungi, and bacteria that can harm close in crops [159]. For example, wild oats (*Avena fatua* L.) can harbor the etiological agents of the powdery mildew in crops such as wheat (*Triticum aestivum* L.), oats, and barley (*Hordeum vulgare* L.); altamisa (*Parthenium hysterophorus* L.) can be a secondary host of the common hairy caterpillar (*Diacrisia obliqua* Walk.); *Cyperus rotundus* can host the root-knot *Meloidogyne graminicola* and, therefore, can contribute to their spreading in the field. Finally, weed infestation may affect fresh and processed products quality such as beer, wine, forage. In this respect, weed residuals may cause accumulation of off-flavors products, or in some cases, can make them harmful to humans and animals. Weeds may also contain high levels of allergens and/or toxic metabolites that, if ingested, can cause asthma, skin rash, and other reactions. Most weed research aims at developing strategies that can reduce the deleterious impact of the interspecific competition between crops and weeds and recent technological advances may further contribute to this scope, while

improving the sustainability of weed control. Worldwide, weed competition causes severe yield reduction in all major crops, such as wheat (23%), soy-bean (37%), rice (37%), maize (40%), cotton (36%), and potato (30%) [150]. Yearly, weeds cause 50% yield losses of corn and soybean productivity in North America. For corn, this equates to a loss of 148 million tons for an economical loss of over $26.7 billion. In Australia, yield loss due to weeds accounts for 2.76 million tons of grain from different plants, including wheat, barley, oats, canola, sorghum, and pulses. The annual global economic loss caused by weeds has been estimated to be more than $100 billion U.S. dollars, despite world-wide annual herbicide sales in the range of $25 billion. In Europe, herbicides are the second most-sold pesticides. They accounted for 35% of all pesticide sales in 2018, overcoming insecticides and acaricides (Figure 4.1)[160].

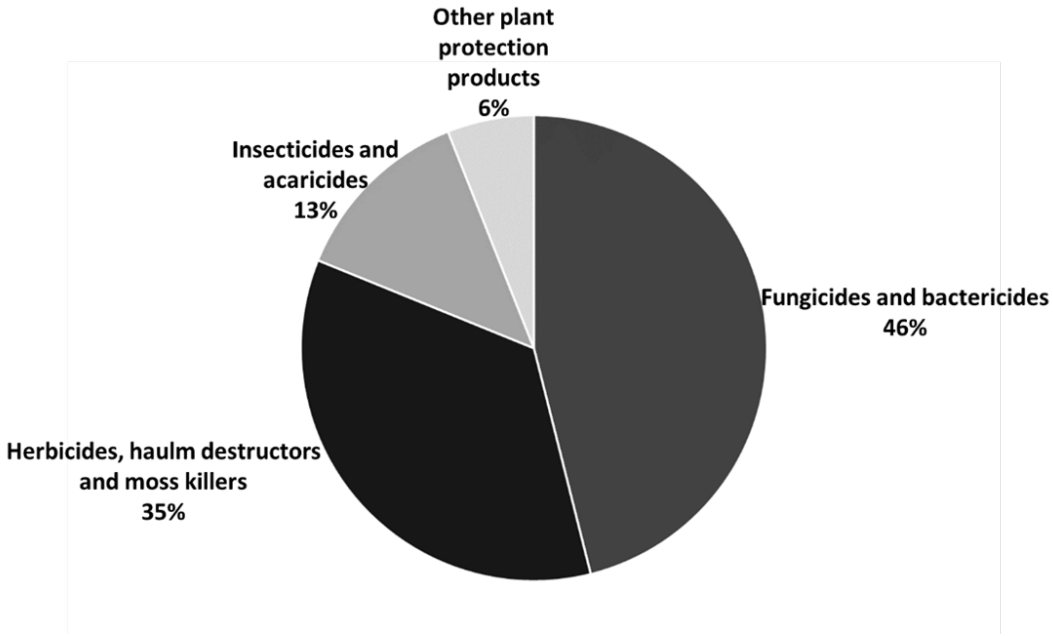


FIGURE 4.1: Percentage (of total volume in kilograms) of pesticide sales by category in Europe in 2018 [160]

4.1.2 Weed management requires an integrated approach

In 2050, the world population will quadruplicate, reaching 9.15 billion people. However, the predicted increase in food demand will be hardly met by the current production system. Also, climate change will be an additional challenge for the human food supply in the near future. Among all the processes affecting crop productivity, weed management will be one of the hardest challenges. Mechanical and chemical weed control has disadvantages that probably will impede them to be effective for future weed management. Mechanical methods are scarcely efficient, and herbicides have a high ecological impact. An approach that minimizes the drawbacks of mechanical and chemical weed control is Integrated Weed Management (IWM). IWM combines chemical, biological, mechanical, and/or crop management methods, and

represents a model to improve the efficiency and sustainability of weed control [152]. In contrast to traditional methods, IWM integrates several agro-ecological aspects such as the role of conservation tillage and crop rotation on weeds seed bank dynamics, the ability to forecast the critical period of weed interference and their competition with crops, and the specific critical levels of crops/weeds interaction. Therefore, an effective IWM must rely on a thorough knowledge of crop-weeds competition dynamics, which currently represents one of the most active research areas in weed science.

4.1.3 New technologies for site-specific weed management

Precision agriculture relies on technologies that combine sensors, information systems, and informed management to optimize crop productivity and to reduce the environmental impact [161]. Nowadays, precision agriculture has a broad range of applications and it is employed in different agricultural contexts including pests control [162], fertilization, irrigation [163], sowing [164] and harvesting [165]. Precision agriculture can be effectively applied to IWM also. In the last decade, precision agriculture has rapidly advanced because of technological innovations in the areas of sensors, computer hardware, nanotechnology, unmanned vehicles systems and robots [166] that may allow for specific identification of weeds that are present in the field. Unmanned aerial vehicles (UAV) are one of the most successful technologies applied in precision agriculture [60]. Unmanned Vehicles systems are mobile Aerial (UAV) or Terrestrial (UTV) platforms that provide numerous advantages for the execution and monitoring of farming activities [167]. UAVs can be highly valuable since they allow for Site-Specific Weed Management (SSWM) (Figure 4.2).

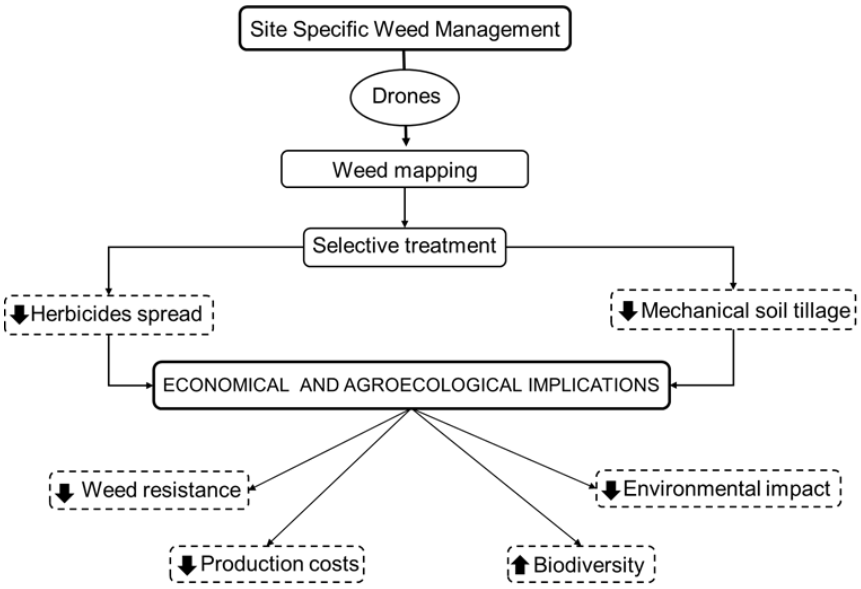


FIGURE 4.2: Site-specific weed management (SSWM) scheme realized by drones and its economical and agro-ecological implications

SSWM is an improved weed management approach for highly efficient and environmentally safe control of weed populations [168], enabling precise and continuous monitoring and mapping of weed infestation. SSWM consents to optimize weed treatments for each specific agronomical situation [169]. The combination of UAVs with advanced cameras and sensors, able to discern specific weeds [170], and GPS technologies, that provide geographical information for field mapping, can help in precisely monitoring large areas in a few minutes. Thanks to more accurate planning of weed management that can increase mechanical methods effectiveness and/or reduce herbicide spread [171], the potential agro-ecological and economic implications of SSWM are remarkable, yielding lower production costs, reducing the onset of weed resistance, improving biodiversity, and containing environmental impacts [172]. The application of UAVs to weed control can, therefore, contribute to improve the sustainability of future agricultural production systems that must comply with an increasing world population [173].

4.1.4 UAVs remote sensing techniques and sensors

UAVs have become a common tool in precision agriculture [174, 175]. Thanks to their affordability, user-friendliness and versatility, UAVs are often the primary choice for fast and precise in situ remote sensing or survey operations. Despite their versatility, these systems may be used for different purposes, depending on the sensors they carry on. Ongoing research is looking at the best solutions to integrate data collected from sensors on UAVs, ground sensors and other data sources for better management of punctual operations in the field, with a particular focus on smart agriculture and big data management [176, 177]. Although UAVs systems do not offer the same territorial coverage as satellites, they offer a spatial and temporal resolution that other systems do not [178, 179]. From an economic point of view, the use of drones requires the investment to buy a UAV system with at least a $0.1 cm/px$ resolution RGB camera, a trained pilot for flight management and post-processing software capabilities. The initial UAV investment is compensated by the repeatability of flights, which increases the frequency of datasets delivered, and the higher resolution compared to other systems [180, 181]. UAVs systems also have further advantages: **(1)** the possibility to collect easily deployable data in real time (excluding post-processing); **(2)** they can be used to survey areas with high level of hazard and/or difficult to reach; **(3)** they allow operators to collect data even with unfavorable weather conditions, such as in very cloudy or foggy days, under which satellite detection systems fail or produce very altered datasets [179]. The most important sensors available as payload are mainly categorized into three classes depending on the spectral length and number they can record:

- RGB (Red, Green, Blue) or VIS (Visible) sensors;
- Multispectral sensors;

- Hyperspectral sensors

The RGB or VIS sensors are the most common and largely available commercial cameras (Table 4.1). Their possible applications have been the focus of most research for years due to their potential and low-cost operational requirements [182, 183]. These sensors are used to calculate vegetation indices such as the Green/Red Vegetation Index (GRVI), Greenness Index (GI) and Excessive Greenness (ExG) with acceptable or high levels of accuracy [184, 185]. Also, RGB sensors have been increasingly used for machine learning techniques in object recognition, phenology, pathologies, and similar purposes. The typical workflow of processing RGB images from UAVs for remote sensing is: **1.** pre-flight planning, **2.** flight and image acquisition, **3.** post-processing and indexes or dataset extrapolation. Phase 1 is critical and essential to collect data of useful quality for the purpose. In the pre-flight planning phase, the parameters to consider are the definition of the study area, the flight altitude, site topography, weather forecast and local regulations for unmanned flights. In phase 2, it is recommended to keep the data flow sufficient to store data and to check if the acquisition platform can acquire the amount of data required. It could be possible to encounter I/O errors due to the inadequacy of the platform with consequent loss of information or abortion of the mission. In phase 3, for RGB sensors, there is no need to perform radiometric calibration, which is the case when using multispectral and hyperspectral sensors. RGB data can be used per se or to create a georeferenced orthomosaic. In this case, the individual images are rectified, georeferenced using GPS data and stitched together to form a single image (orthomosaic) covering the entire study area. Orthomosaics can be generated either with RGB values as they are or after calculating the desired vegetation indices. If RGB images are to be used in machine learning algorithms, the workflow is different [186, 187, 188, 189]. In this case, it is necessary to collect a large dataset of images for the training and testing of the algorithm [190]. This dataset may already be available from third-party sources, such as PlantVillage [191] or PlantDoc [192]. Alternatively, it can be created from scratch if the purpose of the research is not covered by existing datasets [193]. In this case, the acquisition, selection and processing of the images are critical, because the final dataset can affect both the training and the use of the neural network, with risks of producing biased results [194].

TABLE 4.1: RGB cameras and their main specifications

Camera Model	**Sensor type and resolution** [*Mpx*]	**Sensor Format**	**Sensor Size** [*mm*]	**Weight** [*kg*]	**Price (approx.)** [€]
Canon EOS 5d Mark IV	CMOS 30.4	Full Frame	36.0 x 24.0	ca. 1.0	ca. 1000
Nikon D610	CMOS 24.3	Full Frame	36.0 x 24.0	ca. 1.250	ca. 1000
Sony Alpha 7R II	CMOS 42	Full Frame Mirrorless	35.0 x 24.0	ca. 0.6	ca. 1200
Sony Alpha a6300	CMOS 24	Small Frame Mirrorless	23.5 x 15.6	ca. 0.8	ca. 800
Panasonic Lumix DMC GX8	CMOS 20	Small Frame Mirrorless	17.3 x 13	ca. 0.5	ca. 1000
Panasonic Lumix DMC GX80	DLMOS 16	Small Frame Mirrorless	17.3 x 13	ca. 0.5	ca. 500
DJI Phantom 4 Pro *	CMOS 20	Small Frame	13.2 x 8.8	ca. 1.5 (with UAV)	ca. 1500 (with UAV)
DJI Mavic 2 Pro *	CMOS 20	Small Frame	13.2 x 8.8	ca. 1.5 (with UAV)	ca. 1500 (with UAV)

The multispectral sensors are used for a wider range of calculations of vegetation indices as they can rely on a higher number of radiometric bands. A comparison of the most common multispectral sensors, specific for UAV systems, is shown in table 4.2.

TABLE 4.2: Multispectral sensors and their main specifications

Camera Model	Resolution [*Mpx*]	Spectral Bands	Ground Sample Distance [*cm/px*]	Weight [*kg*]	Price (approx.) [€]
Micasense RedEdge-M	1280 x 960 (1.2 Mpx per EO band)	Red, Green, Blue, Near-Infrared, Red Edge	8 (per band) at 120 m AGL	ca. 0.180	ca. 5000
Micasense RedEdge-MX	1280 x 960 (1.2 Mpx per EO band)	Blue, green, red, red edge, near infrared (NIR)	8 (per band) at 120 m AGL	ca. 0.231	ca. 5000
Micasense Altum	2064 x 1544 (3.2 Mpx per EO band) 160 x 120 thermal infrared	EO: Blue, green, red, red edge, near-infrared (NIR) LWIR: thermal infrared 8-14um	5.2cm per pixel (per EO band) at 120m AGL 81cm per pixel (thermal) at 120m	ca. 0.405	ca. 6000
TertaCam MCAW 6	1.3	6 user selectable narrow bands (450 µm to 1000 µm)	-	ca. 0.550	ca. 17000
TetraCam ADC Lite	3.2	Green, Red, Near-Infrared (NIR)	-	ca. 0.2	ca. 3000
TetraCam ADC Micro	3.2	Green, Red, Near-Infrared (NIR)	-	ca. 0.09	ca. 3000
Parrot Sequoia+	1.2	Blue, Green, Red, Red Edge, Near-Infrared (NIR)	-	ca. 0.7	ca. 5000

With multispectral sensors, the range of vegetation indices that can be monitored is considerably extended compared to those that can be calculated with only three RGB bands. Moreover, the workflow has minor variations. For these sensors, in phase 1, the radiometric calibration and atmospheric correction phases are strictly required. Many multispectral sensors, such as the Micasense RedEdge series or the Parrot Sequoia + , have downwelling irradiance sensors and a calibrated reflectance panel to address some of the requirements for radiometric calibration 5.1. Due to a lower resolution of the sensors compared to RGB ones, a lower flight altitude and an adequate horizontal and vertical overlap of recorded images must be taken into account to obtain an adequate ground resolution for the surveyed objective and to avoid missing data. In phase 2, having a higher number of radiometric bands to record, the dataflow will be higher so is critical to avoid I/O errors, missing data or mission failures [195]. Due to multi-lenses nature of the sensors in phase 3, the data collected suffer from the parallax problem. As a consequence, images have to be rectified, georeferenced and must be stacked to generate a single image with different radiometric levels, and calibrated with the downwelling irradiance sensors data acquired during the flight [196]. After this procedure, it is possible to generate a multispectral orthomosaic and then calculate the requested indexes [197]. Multispectral images are also used in machine learning applications taking into account the multi-camera nature of sensors and the different bands recorded. Thanks to the availability of a higher number of radiometric bands, the machine learning algorithms can be extended to not-visible recognition such as early stage plant disease, field quality assessment, soil water content, and more [197].

The hyperspectral sensors can record hundreds to thousands of narrow radiometric bands, usually in visible and infrared ranges. To deal with hyperspectral applications, the choice of number and radiometric range of bands is critical. Each band or combination of bands, being very narrow, can detect a specific field characteristic. Each hyperspectral sensor can detect only a certain number of bands, so the aim of survey must be very clear to choose the right sensor. Although hyperspectral sensors have decreased in price in recent years, they are still an important starting investment since they are much more expensive than RGB and multispectral sensors.

In addition, they are heavier and bigger than other sensors, often making their use on UAV systems difficult and/or excessively onerous in terms of payload. Some of most used hyperspectral sensors in UAVs application and their main characteristics are shown in table 4.3.

TABLE 4.3: Hyperspectral sensors and their main characteristics

Camera Model	**Lens**	**Spectral Range [*μm*]**	**Spectral Bands [*Number and μm*]**	**Weight [*kg*]**	**Price (approx.) [€]**
CUBERT	Snapshot + PAN	450 – 995	125 (8 μm)	ca. 0.5	Ca. 50000
Cornirg microHSI 410 SHARK	CCD/CMOS	400 – 1000	300 (2 μm)	ca. 0.7	-
Rikola Ltd. hyperspectral camera	CMOS	500 – 900	40 (10 μm)	ca. 0.6	ca. 40000
Specim-AISA KESTREL16	Push-broom	600 – 1640	350 (3 – 8 μm)	ca. 2.5	-
Headwall Photonics Micro-hyperspec X-series NIR	InGaAs	900 – 1700	62 (12.9 μm)	ca. 1.1	-

In this case, the workflow for radiometric calibration is more complex compared to other sensors. Some calibration methods needed for these sensors are derived from manned aircraft hyperspectral platforms, based on artificial targets to assess data quality, to correct radiance, and to generate a high-quality reflectance data-cube [198]. In phase 1, the planning must also be carried out in time and not only in space because, in addition to the spectrometric resolution, hyperspectral sensors have a temporal resolution due to the different acquisition method. In phase 2, it should be considered that both images' size and data flow are bigger than multispectral/RGB images. Moreover, these sensors may acquire a large amount of data, but the payload limitations of UAVs may not allow the transport of adequate file storage systems. Phase 3 for hyperspectral images is critical: quality assessment is one of the critical issues of hyperspectral data and some problems associated with the quality of the images have not been completely overcome. Among those, the stability of the sensor itself (due to the nature of UAV platforms) and the vibrations involved can comprise a good calibration of the sensor. Subsequently, on post-processed data, it is possible to calculate narrowband indices such as chlorophyll absorption ratio index (CARI), greenness index (GI), greenness vegetation index (GVI), modified chlorophyll absorption ratio index (MCARI), modified normalized difference vegetation index (MNDVI), simple ratio (SR), transformed chlorophyll absorption ratio index (TCARI), triangular vegetation index (TVI), modified vegetation stress ratio (MVSR), modified soil-adjusted vegetation index (MSAVI) and photochemical reflectance index (PRI) [199].

4.1.5 Applications of UAVs to weed management

UAVs are ideal to identify weed patches. The main advantages of UAVs compared to UTVs are the shorter monitoring/surveying time they require and optimal control in the presence of obstacles, which is critical when working between crop rows [200] In a few minutes, UAVs can cover many hectares flying over the field, thus providing the photographic material for weed patches identification. These images are processed via deep neural network, convolutional neural network, and object-based

image analysis [201, 202]. Based on a systematic review of the literature concerning weed identification by UAVs, it can be concluded that mainly three types of cameras are used for weed patches identification: RGB, multispectral and hyperspectral cameras (Table 4.4).

TABLE 4.4: Weed patches identification by different types of camera (multispectral, RGB, hyperspectral)

Crop	Weed (common name)	Weed (scientific name)	Type of camera	Main results
	- Palmer amaranth	- *Amaranthus palmeri*	Hyperspectral camera	Discriminate glyphosate-resistant from glyphosate-sensitive weeds
	- Spotted knapweed	- *Centaurea maculosa*	Hyperspectral camera	Detection invasive species affecting forests, rangelands, and pastures
	-Babysbreath	- *Gypsophila paniculata*		
	- Bunchgrass	- *Phalaris minor*		
	- Egyptian crowfoot grass	- *Dactyloctenium aegyptium*	RGB camera	Identify different weeds
	- False amaranth	- *Digera arvensis*		
	- Awnless barnyard grass	- *Echinochloa colona*		
	- Ragwort	- *Jacobaea vulgaris (Senecio jacobaea)*	Multispectral camera	Discriminate weeds in pastures
	-Buffel Grass	- *Cenchrus ciliaris*	RGB camera	Discriminate two different weeds
	-Spinifex	- *Triodia sp.*		
- Beta vulgaris **- Zea mays** **- Hordeum vulgare** **- Lens esculenta** **- Pisum sativum** **- Phaseolus vulgaris** **- Carthamus tinctorius** **- Cicer arietinum** **- Triticum spp.** **- Triticosecale**	- Kochia - Marestail - Common lambsquarters	- *Bassia scoparia* - *Conyza canadensis* - *Chenopodium album*	Hyperspectral camera	Discriminate glyphosate and dicamba resistant genotypes from sensitive genotypes
			RGB camera	Comparison of cereal genotypes
Beta vulgaris	Weeds		Multispectral camera	Discriminate crop vs weeds
Beta vulgaris	Weeds		Multispectral camera	Discriminate crop vs weeds
Beta vulgaris	- Thistle	- *Cirsium arvense*	Multispectral camera	Discriminate crop vs weeds
Beta vulgaris	- Thistle - Wild buckwheat - Ryegrass	- *Cirsium arvense* - *Fallopia convolvulus* - *Lolium multiflorum*	Multispectral camera	Discriminate crop vs weeds
Beta vulgaris	- Thistle	- *Cirsium arvense*	Multispectral camera	Discriminate crop vs weeds
Cicer arietinum	Weeds		Hyperspectral camera	Discriminate crop vs weeds
Glycine max	- Palmer amaranth - Barnyardgrass - Large crabgrass	- *Amaranthus palmeri* - *Echinochloa crus-galli* - *Digitaria sanguinalis*	- RGB camera - Multispectral camera	Assessment of crop injury from dicamba
Heliathus annuus	- Pigweed - Mustard - Bindweed - Lambsquarters	- *Amaranthus blitoides* - *Sinapis arvensis* - *Convolvulus arvensis L* - *Chenopodium album L*	- RGB camera - Multispectral camera	Discriminate crop vs weeds
Hordeum vulgare	- Thistle - Coltsfoot	- *Cirsium arvense* - *Tussilago farfara*	RGB camera	Discriminate crop vs weeds
Hordeum vulgare	- Thistle	- *Cirsium arvense*	RGB camera	Discriminate crop vs weeds
Hordeum vulgare	- Thistle	- *Cirsium arvense*	RGB camera	Discriminate crop vs weeds
Lactuca sativa	- Common groundsel - Shepherd's purse - Sow thistle	- *Senecio vulgaris* - *Capsella bursa pastoris* - *Sonchus spp.*	Multispectral camera	Discriminate crops vs weeds
Sorghum spp.	- Amaranth - Pigweed - Barnyard grass - Mallow - Nut grass - Fat Hen	- *Amaranthus macrocarpus* - *Portulaca oleracea* - *Echinochloa crus galli* -*E. colona* -*Malva spp.* - *Cyperus rotundus* - *Chenopodium album*	Hyperspectral camera	Discriminate crop vs weeds
Triticum durum	- Wild oat - Canarygrass - Ryegrass	- *Avena sterilis* - *Phalaris canariensis* - *Lolium rigidum*	Multispectral camera	Discriminate crop vs weeds
Triticum durum	- Wild oat - Canarygrass - Ryegrass	- *Avena fatua* - *Phalaris canariensis* - *Lolium rigidum*	- Hyperspectral camera -Multispectral camera	Discriminate crop vs weeds
Triticum sp.	- Thistle	- *Cirsium arvense*	RGB camera	Discriminate crop vs weeds
Triticum spp.	Weeds		Hyperspectral camera	Discriminate crop vs weeds
Vitis vinifera	- Bermuda grass	- *Cynodon dactylon*	RGB camera	Discriminate crop vs weeds
Zea mays	Weeds		Multispectral camera	Discriminate crop vs weeds
Zea mays	- Common lambsquarters - Thistle	- *Chenopodium album* - *Cirsium arvense*	Multispectral camera	Discriminate monocotyledons (crops) vs dicotyledons (weeds)
Zea mays	- Common lambsquarters - Thistle	- *Chenopodium album* - *Cirsium arvense*	Multispectral camera	Discriminate crop vs weeds
Zea mays	- Mat amaranth - Johnsongrass	- *Amaranthus blitoides* - *Sorghum halepense*	Multispectral camera	Discriminate crop vs weeds

These cameras are very similar in terms of information obtained for the purpose of weeds identification. Indeed, the three camera types can recognize weed patches with good accuracy depending on flying altitude, camera resolution and UAV used. UAVs have been mainly tested on important crops such as *Triticum spp.*, *Hordeum vulgare*, *Beta vulgaris*, *Zea mays* [203]. These are among the most cultivated crops

worldwide and are highly susceptible to weed competition especially in early phenological stages. In these crops, it was possible to identify several dicotyledonous weeds including Amaranthus palmeri, *Chenopodium album* and *Cirsium arvense* [204, 205], as well as different monocotyledonous such as *Phalaris spp.*, *Avena spp.* and *Lolium spp.* [206, 207]. These weed species are wide-spread globally and can be a serious threat to different crops. Therefore, the combined use of UAVs and image processing technologies may contribute to effectively control different weed species interfering with the crops with relevant environmental benefits [208].

4.1.6 Conclusions

The use of UAVs and machine learning techniques allow for the identification of weed patches in a cultivated field with accuracy and can improve weed management sustainability [202]. Weed patches identification by UAVs can facilitate integrated weed management (IWM), reducing both the selection pressure vs herbicide-resistant weeds and herbicides diffusion in the environment [171]. Recent research has shown that new technologies are able to discern single weed species in open fields [170, 207]. If integrated with weed management planning, this information gathered via remote imaging analysis can contribute to sustainably improve weed management. In addition, imaging analysis can help in the study of weed dynamics in the field, as well as their interaction with the crop, which both represent a necessary step to define new strategies for weed management based on interspecific crop–weed interactions [209, 210, 211]. Recent studies demonstrate that some weed communities are actually not detrimental to crop yield and quality [209, 210]. In winter wheat cultivation, a highly diversified weed community caused lower yield losses than a less diversified one [211]. In soybean, through a combination of field experiments in which weed species were manipulated in composition and abundance, it has been shown that increasing levels of weed competition resulted in an increase in seed protein content without impairing yield. Most likely, the integration of known and emerging technologies in this field will greatly improve the sustainability of weed control, following the SSWM approach. By image analysis, different machine learning techniques will be able to provide a reliable overview of the level and type of infestation. Specific algorithms can be trained to manage weeds removal by Autonomous Weeding Robot (AWR), via herbicide spray or mechanical means, as showed in figure 4.3 as proof of concept [212].

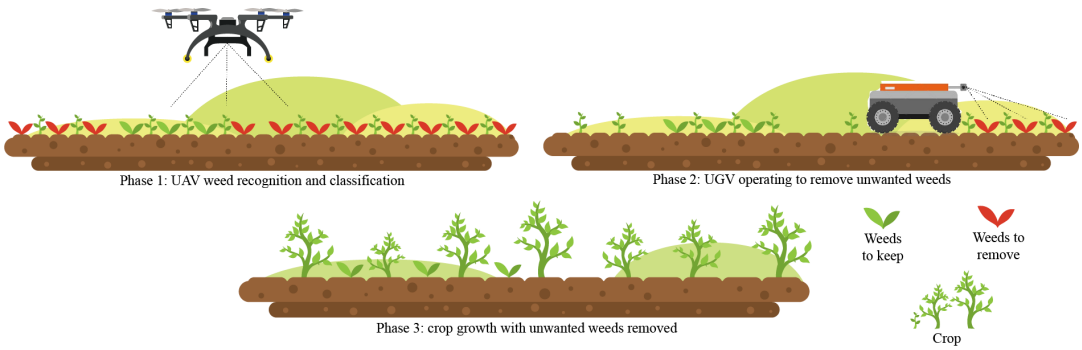


FIGURE 4.3: A proof of concept of an integrated automatic weed management system

Also, the creation of a specific weed images dataset is crucial to achieve this goal. This approach must necessarily rely on a dataset of photographs taken in dedicated experimental fields, labeled in extended COCO/POCO (Common Objects in COntext/ Plant Objects in COntext) format [194] and integrated with images from Plant-Village dataset [191] or other existing ones. New insights on weed population dynamics and their competition with crops are needed in order to extend this approach to real agricultural contexts, so as to specifically recognize and eliminate only harmful weed species. The overall objective is to overcome the consequences of biological vacuum around the crop, which has been proved to be highly impacting for both biotic and the abiotic components of the environment [213, 214], with long-term consequences on human safety on earth.

4.2 Optimizing Crop Segmentation in Heterogeneous Agriculture Scenario across Phenology of *Brassica oleracea var. botrytis*

4.2.1 Introduction

Weeds, which are defined as undesirable plant species, grow simultaneously along the crops and consume the nutrients which are mandated only for crop consumption. This inhibits the crop growth and harbors threat in terms of viral crop diseases and problematic insects. They are mostly damaging to crop yield if they have advantages over the crop in some or the other way. Mapping and removal of weeds in early stages of weed emergence will not only be relatively less complicated but also less time-and-cost consuming over later stages of the crop where there is high spectral overlap and heterogeneity. Therefore, there's a need for mechanism which facilitates identification of crops and weeds in early stages so that irrespective of temporal change, crops and/or weeds can be segmented and classified across all phenological stages. There are various publications dealing in discrimination of crops and weeds in digital images, over parameters of spectra, geometry, texture, and height (see 4.1. Although commercially some solutions are oriented at early

stages of weed growth, there's a need for less computationally expensive solutions which can facilitate and augment segmentation. This produces a cascading requirement for tools which should be capable of periodical monitoring of crops for precise intervention for weed removal. If spraying of weedicides could be limited to just the weed-affected areas, the ecological harm in terms of soil contamination and overspraying onto crop regions can be reduced. This can be achieved when the weed information in aerial images is supported by geolocation information at millimeter level accuracy. The assessments in this subsection showed that crop and weed segmentation through use of machine learning and deep learning algorithms like ResNet, DenseNet, U-Net, DeepCluster DeepLab and VGGNet had limitations in terms of handling of spectral overlap and performance in heterogenous environments.

4.2.2 Materials and Methods

The study in this subsection is focused on facilitating crop segmentation across early stages of cauliflower in a heterogeneous environment where there are varying lighting conditions, different soil moisture levels, occlusion and infestation by weeds. The sensor in use is DJI Phantom IV Pro for capturing true color (RGB) images and Micasense RedEdge-M for capturing multi-spectral images of cauliflower crop field at different time intervals. The study area is a cauliflower field located in Department of Agricultural Sciences, University of Naples Federico II, Portici, Italy. The classes of interest in the imagery are weeds and cauliflower leaves, The analysis over these images was done by making use of several band ratios such as Normalized Difference Vegetation Index (NDVI), Modified Soil Adjusted Vegetation Index (MSAVI2), some image super resolution techniques and morphological operations for optimization of segmentation outputs. The preliminary results have crop segmentation masks as the end output. The subsequent step would be automatic annotation. Current methods require annotating images in a manual fashion via image labeling software/tools such as CVAT or LabelStudio which is time consuming and tedious. The aim is therefore to facilitate a tool that improves the quality of the image in terms of level of details and crop characters in presence of weeds. There is also a requirement to understand the spectral overlap among weeds and cauliflower leaves in order to quantify the correlation among both. The final goal is to improve the recognition of weeds and ameliorate the classification process amongst underlying species of weeds. Once the crops are segmented in the early stages of weed growth, these results can be a prerequisite for training and annotations. Mapping weed affected areas for mechanized-spraying of herbicides through drones (UAVs) and/or rovers (UTVs). Not only this will provide a huge relief from ecological perspective but also beneficial for any project from economical point of view.

Flight missions were scheduled at a one-week time interval in order to appreciate the phenological growth of weeds. Two flights were conducted each week: one with an RGB sensor and the other with a multispectral sensor (MicaSense RedEdge-M)

with a mission that ensured an overlap between successive snapshots of at least 75%. An example flight plan with the relative positions of the individual shots is shown in figure 4.4.

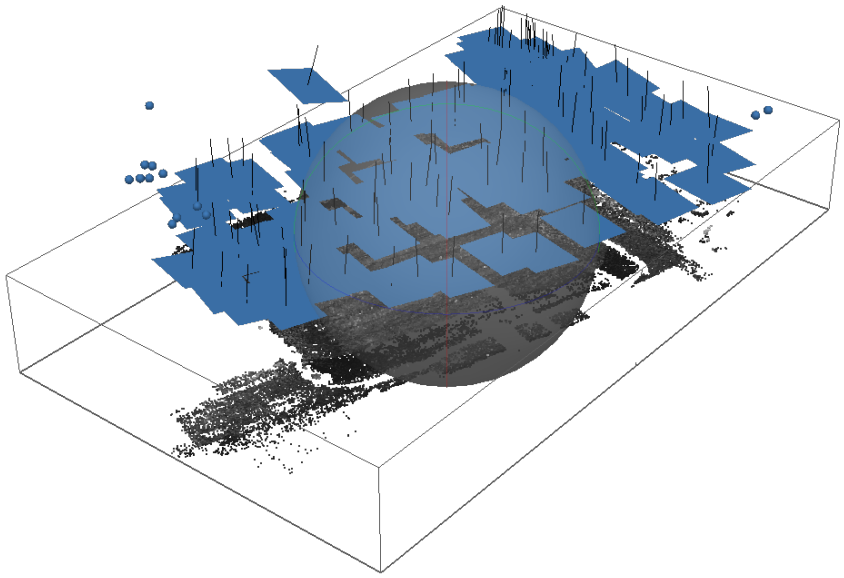


FIGURE 4.4: Flight plan of a multispectral survey mission

4.2.3 Results and Discussion

First results were obtained by applying object based segmentation algorithms. A first approach was to segment the crop and then exclude it from the subsequent process, in order to segment only the weeds while excluding the crop. The results of this first approach can be seen in figure 4.5, where different segmentation techniques based on different algorithms are shown. figure 4.5d shows a first segmentation of the weeds obtained with the elimination of the cultivation thanks to the segmentation of the first phase, as explained above.

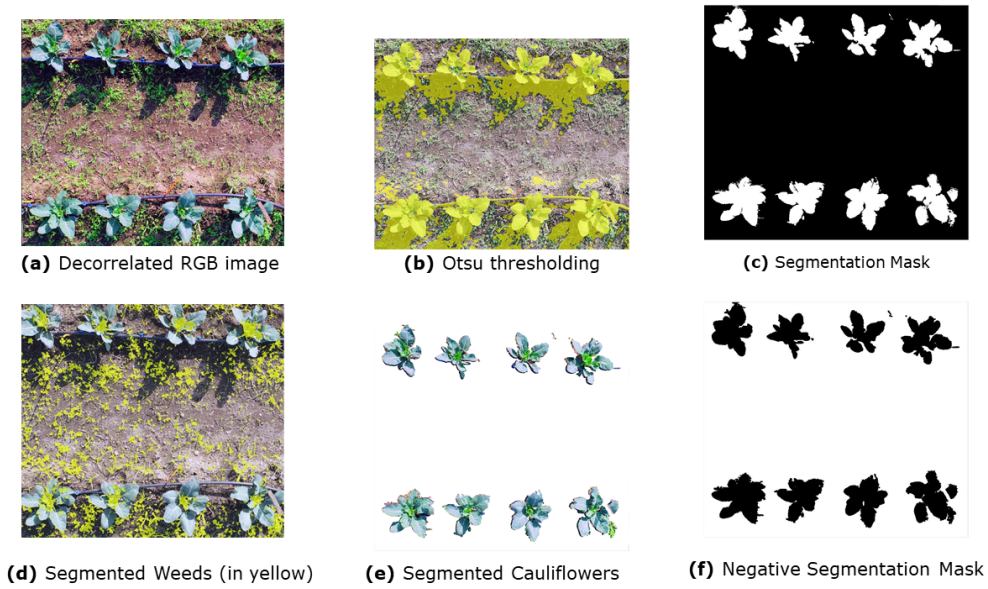


FIGURE 4.5: First results with segmentation algorithms

Subsequently, the algorithm was tuned to be able to allow direct weed segmentation. The processing area was confined to the intra-row and inter-row space of the crop, as shown in figure 4.6.

FIGURE 4.6: Area of interest for segmentation algorithm

The key steps in this approach were as follows:

- UAV image sequence;
- Subset creation;
- Bicubic SR at interpolation factor 4 (optional);
- MSAVI2 (Modified soil adjusted vegetation index);
- Otsu thresholding;

- Mask inversion;
- Morphological erosion by radius > 0 and radius < 3 for weeds And radius > 7 and radius < 10 for cauliflowers.

The results can be seen in figure 4.7 where it can be seen that, starting from a pure RGB image (figure 4.7a), a first segmentation results unsatisfactory (figure 4.7b), while using a multispectral index that allows the removal of the soil and isolate the weeds (in the specific case MSAVI2), the segmentation of the weeds results better and satisfactory (figure 4.7c).

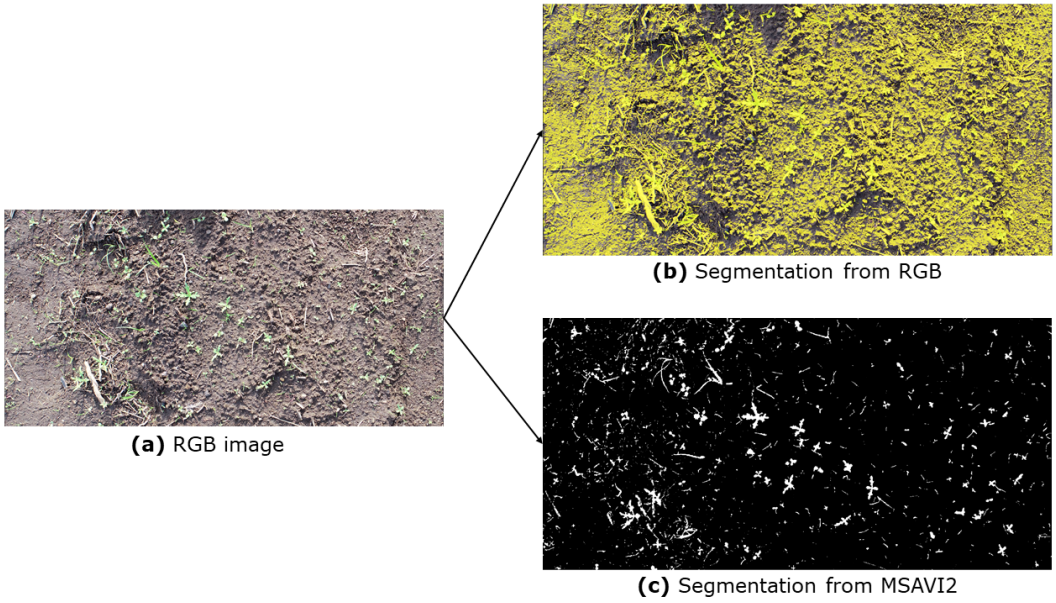


FIGURE 4.7: Weed segmentation in area of interest

Preliminary results show that segmentation even of weeds in an early phenological state is possible. However, the segmentation algorithm needs to be improved in order to handle the various time phases of weed growth during which the shape, size, and morphological characteristics of weeds change.

4.2.4 Conclusions

By creating automated segmentation masks, it is possible to speed up, or even automate, the process of labeling images in order to use the datasets to train an artificial intelligence to recognize weeds using Deep Learning algorithms. The results displayed above show promising approaches towards this purpose. The first approach of segmenting the crop and then excluding it and segmenting only the weeds turned out to be unsatisfactory in contrast to the approach of directly segmenting the weeds using pure RGB source images or via multispectral indices. Further studies are underway to refine the segmentation algorithms and automate the process.

4.3 Scientific work produced

[215] - Marco Esposito, Mariano Crimaldi, Valerio Cirillo, Fabrizio Sarghini, Albino Maggio. Drone and sensor technology for sustainable weed management: a review. In: Chemical and Biological Technologies in Agriculture. 8.1 (2021). Publisher: SpringerOpen, pp. 1–11.
[216] - Marco Esposito, Valerio Cirillo, Mariano Crimaldi, Fabrizio Sarghini, Albino Maggio. Sustainable Weed Management: Modern Approaches and New Perspectives. In: Sustainable management of cropping systems. Bari,2020, p. 24.

Shubham Rana, Salvatore Gerbino, Petronia Carillo, Marco Esposito, Mariano Crimaldi, Valerio Cirillo, Albino Maggio, Fabrizio Sarghini. Optimizing Crop Segmentation in Heterogeneous Agriculture Scenario across Phenology of *Brassica oleracea var. botrytis*. In: Living Planet Symposium. 23-27 May 2022. Bonn, Germany.

Chapter 5

Digital agriculture in ecohydraulics applications

5.1 Bulk Drag Predictions of Riparian *Arundo donax* Stands through UAV-Acquired Multispectral Images

5.1.1 Introduction

The presence of aquatic vegetation inside vegetated open channels has a strong impact on flood hazards in urban areas [217, 218, 219]. The identification of the riparian vegetation traits affecting flow resistance is one of the most relevant topics of both modeling and experimental analyses in ecohydraulics [220, 221]. Riparian plants behave differently depending on their bio-mechanical and morphometric traits, and on their submergence. According to Västilä and Järvelä [220], when flexural rigidity of riparian plant's stems is very far from negligible, flow resistance associated with weed riparian vegetation species can be computed by exploiting Leaf Area Index (LAI). to estimate the so-called bulk drag coefficient (C_D)[222, 223]. The implications of the interaction between flow and vegetation in vegetated water bodies on their main fluid dynamic features have been largely investigated in many experimental and numerical analyses by schematizing riparian plants as natural-like elements [224, 225, 226, 227]. Among other methods associated with remote sensing data, digital processing of images acquired through Unmanned Aerial Vehicles (UAV) represents a promising tool for mapping the most relevant riparian vegetation parameters within vegetated water bodies. UAV-based multi and hyperspectral images have been largely applied in many forestry and precision agriculture studies [228, 229, 230, 231]. However, a methodology for predicting the main hydrodynamic features of real vegetated water bodies based on these approaches is still an open research window. Indeed, most previous studies in real-scale riparian vegetation and hydrodynamic conditions have analyzed riparian plants by directly collecting them from the field, as reported by Västilä and Järvelä [220]. In this chapter is presented one of the first efforts in estimating and validating bulk drag coefficients (C_D) induced by 9–10m high *Arundo donax* stands, a very common

riparian species [232], covering an abandoned drainage channel, based on vegetative flow resistance model proposed by Västilä and Järvelä for rigid plants [220] through the digital processing of UAV-acquired multispectral images. As shown by Etminan et al. [233], when dealing with real vegetated water bodies, vortices and mixing production associated with riparian vegetation stands depends on the only bulk drag coefficients (C_D), which consequently affects the environmental quality of both terrestrial and aquatic ecosystems within. Thus, to predict the effects of riparian vegetation stands morphometry on the hydrodynamic behavior of real vegetated streams, the knowledge of real-scale bulk drag coefficients (C_D) is essential. First, the study case is presented in terms of ecohydraulic and phenological conditions of the examined riparian vegetation species. Streamwise velocity components and ground-based Leaf Area Index (LAI) measurements were carried out at 30 cross-sections uniformly spaced along the examined vegetated drainage channel by means of a propeller-type OTT® C31 Universal Current Meter (OTT HydroMet - Kempten, Germany) and LI-COR® LAI2000 Plant Canopy Analyzer (LI-COR Inc. - Lincoln, Nebraska USA) device, respectively. The ground-based LAI measurements of the portion of *Arundo donax* stands effectively interacting with water flow (hereinafter indicated as LAI^*) were then correlated to UAV-based Normalized Difference Vegetation Index ($NDVI_{UAV}$) maps of the examined riparian stands to obtain LAI^* maps of the whole channel, whose values are hereinafter indicated as LAI^*_{UAV}. In detail, 10 ground-based LAI^* measurements were randomly taken across each of the 30 examined cross-sections, and the $NDVI_{UAV}$ values corresponding to the pixel located at the same measuring points were then compared to them. It was demonstrated that the uncertainties associated with LAI^*_{UAV*} of the examined $9-10m$ high Arundo donax stands based on digital processing of multispectral images do not heavily affect the distribution of (C_D) predictions along the examined vegetated drainage channel, being NDVI a very robust predictor of LAI^*.

5.1.2 Materials and Methods

Study Area

The study area examined in the present study is a $115.5m$ long and $5.50m$ wide abandoned vegetated drainage channel, with a slope approximately equal to 0.5%, colonized by $9-10m$ high *Arundo donax* stands, most known as giant reed, located in Nola (Campania, Southern Italy), as displayed in figure 5.1. Three experimental field campaigns were carried out in the study area, two of them devoted respectively to flow velocity measurements and ground-based Leaf Area Index measurements of *Arundo donax* stands' portion effectively interacting with flow (LAI^*) at 30 channel's cross-sections, whilst the third was devoted to Normalized Difference Vege-tation Index (NDVI) mapping derived by UAV-acquired multispectral images of the examined riparian plants.

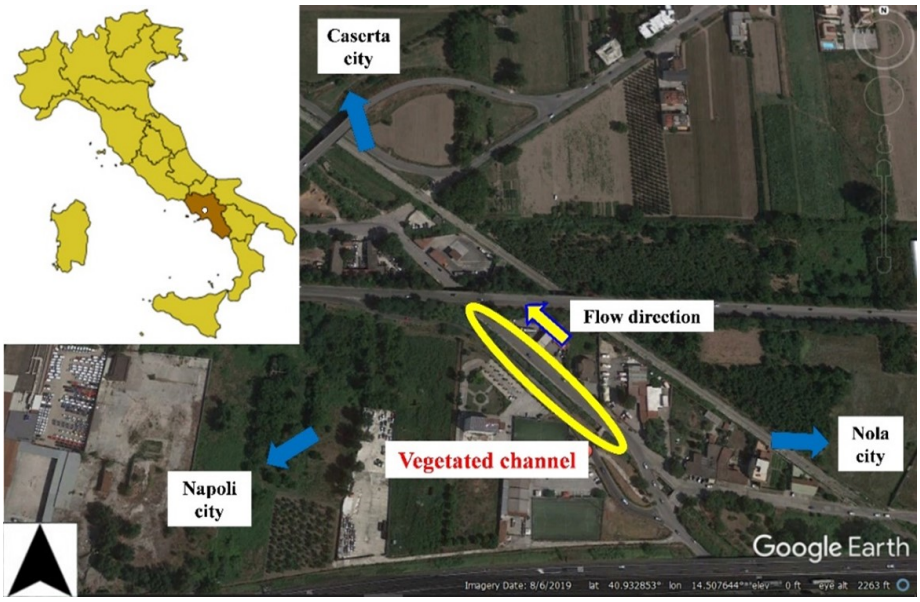


FIGURE 5.1: Study area overview: the yellow ellipse indicates the vegetated drainage channel fully covered by $9-10m$ high *Arundo donax* stands. The yellow arrow indicates the flow direction

The examined channel serves a densely urbanized area, representing both a key infrastructure for mitigating pluvial and flash floods and offers important ecosystem services. It is uniformly $2m$ deep and thus the $9-10m$ high *Arundo donax* stands were emergent. Figure 5.2a gives a view of the examined drainage channel, while figure 5.2b and figure 5.2c show respectively its upstream and downstream cross-sections.

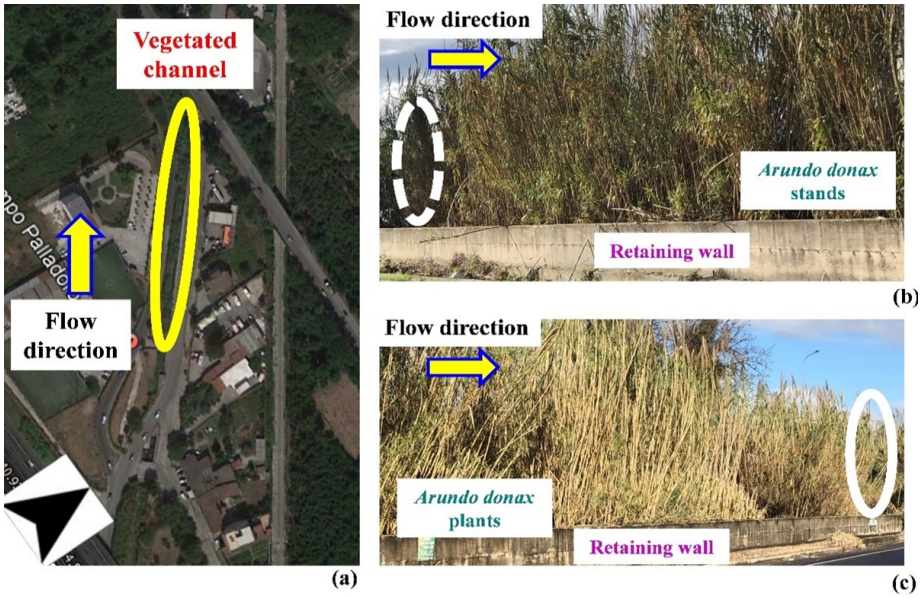


FIGURE 5.2: **(a)** Aerial view of the vegetated drainage channel fully covered by $9-10m$ high *Arundo donax* stands, indicated by the yellow ellipse, and drainage channel's retaining wall. **(b)** Upstream and **(c)** Downstream cross-sections of the vegetated drainage channel, respectively indicated by dashed and continuous white ellipses. The yellow arrow indicates the flow direction

As depicted in figure 5.3, LAI^* and flow velocity measurements were carried out at 30 cross-sections, uniformly distributed along the vegetated drainage channel examined in the present study, located 3.85m apart. In detail, 10 LAI^* measurements were randomly carried out at each cross-section for a total of 300 data, while the streamwise velocity measurements were performed at 27 measuring points belonging to an experimental measuring grid arranged to calculate the flow average velocity U. It was then possible to estimate a discharge of $0.14 m^3 s^{-1}$.

FIGURE 5.3: Aerial view of the 30 cross-sections of the vegetated drainage channel fully covered by $9 - 10m$ high *Arundo donax*, located 3.85m apart. The yellow arrow indicates the flow direction

The three experimental setups adopted here to perform streamwise velocity components u, LAI^*, and NDVI measurements are described in the next subsections.

Experimental Hydrodynamic and Riparian Vegetation Measurements

Hydrodynamics of vegetated streams is highly affected by the different responses of the natural elements involved in the real-scale interaction between water flow and riparian plants to water flow. As shown in figure 5.4, rigid and emergent reed, or giant reed stands, covering vegetated open channels are subjected to two forces according to the well-known cantilever beam theory: weight force W, proportional to gravity acceleration g and drag force F_D, proportional to flow average velocity U.

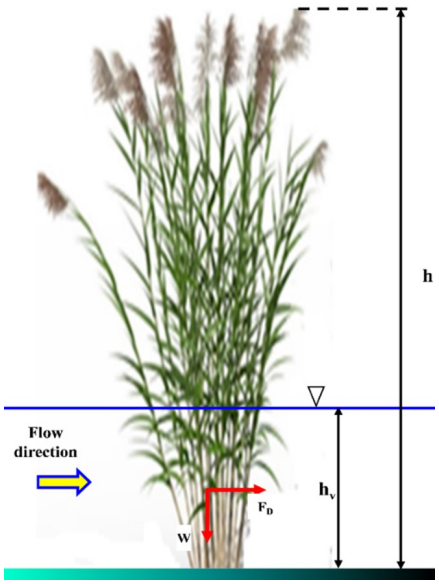


FIGURE 5.4: Hydrodynamic scheme of rigid emergent *Arundo donax* stands interacting with water flow in vegetated streams: h and h_v are the water level and the height of the stand, respectively. The yellow arrow indicates the flow direction. Adapted from Lama and Chirico [223]

Under these physical assumptions, flow average velocity U can be expressed as follows [223]:

$$U = \sqrt{\frac{2g}{\overline{C_D} LAI^*} RJ} \tag{5.1}$$

where $\overline{C_D}$ is the so-defined bulk drag coefficient [220] to which corresponds the flow average velocity U of each *Arundo donax* stand analyzed here, LAI^* is the Leaf Area Index of the portion of *Arundo donax* stands effectively interacting with flow, R is the hydraulic radius defined by the ratio between flow area wetted perimeter at each cross-section, and J is the slope of the energy line, equal to the channel's bed longitudinal slope under the hypothesis of uniform flow conditions. Thus, by applying the model proposed by Västilä and Järvelä [220], it is possible to compute the corresponding values of bulk drag coefficients $\overline{C_D}$ as follows:

$$\overline{C_D} = \frac{2g}{U^2 LAI^*} RJ \tag{5.2}$$

Streamwise velocity components u (ms^{-1}) were measured at each of the 30 channel's cross-sections through a propeller-type OTT® C31 Universal Current Meter, as depicted in figure 5.5a. The experimental measuring grid was composed of 27 points spaced $0.50m$ in both vertical and spanwise (or horizontal) directions, as shown in figure reffig:measuringgridb. Flow area and wetted perimeter of each cross-sections were assessed by considering the real channel's bed topography. Flow average velocities U were then computed once the experimental cross-sectional u distribution at each examined channel's cross-section was known.

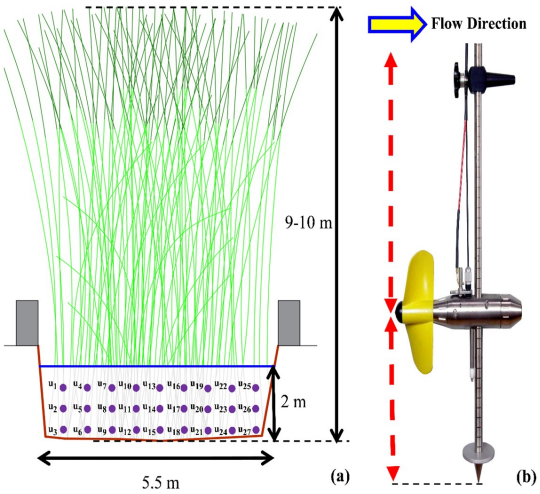


FIGURE 5.5: **(a)** Scheme of the measuring grid composed of 27 points at each cross-section (violet measuring points) employed for the experimental velocity measurements. **(b)** Propeller-type OTT® C31 Universal Current Meter device, sliding on a vertical round wading rod (red bidirectional dashed arrows). The yellow arrow indicates the flow direction

LAI analysis is representative of riparian vegetation growth and health, widely employed in the analysis of climate change effects on flooding hazards, as reported in many works on the protection and improvement of ecological services in both natural and manmade environments [199, 234, 184]. In the present study, a total of 300 LAI^* measurements were performed at 30 cross-sections, spaced 3.85m along the vegetated drainage channel under diffusive sky conditions to reduce the effect of sunlight fluctuations. As displayed in figure 5.6a, ten LAI^* measurements were randomly retrieved at each cross-section and then used for further statistical analyses, according to an experimental methodology introduced by Lama et al. [235]. In their research, the authors analytically quantified the uncertainty in indirect LAI measurements and its impact on the accuracy of flow velocity estimations based on the vegetative flow resistance model proposed by Västilä and Järvelä [220]. After a preliminary phase of signal calibration indicated by figure 5.6b, the 150° wide optical sensor of LI-COR® LAI2000 Plant Canopy Analyzer device measured the rate of sunlight transmitted through the *Arundo donax* canopy corresponding to the portion of vegetation going from the top of the channel's retaining wall (position G1 in figure 5.6a) to water table (position G2 in figure 5.6a), indicated as Δ^*. It was computed as the difference between the ground-based LAI measurements carried out at position G1 and those per-formed at position G2. An example of ground-based LAI measurements carried out at position G1 is reported in figure 5.6c.

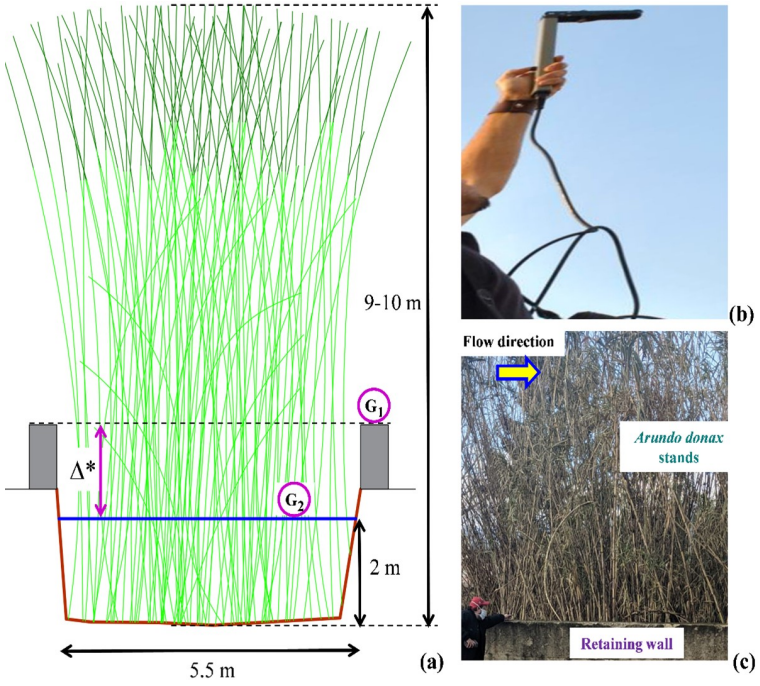


FIGURE 5.6: **(a)** Experimental scheme of LAI^* measurements: G1 and G2 are respectively the measuring positions located above the channel's retaining wall and above water level (purple circles), while Δ^* is the portion of *Arundo donax* effectively interacting with flow (purple line), to which corresponds LAI^*. **(b)** Sunlight calibration of LI-COR® LAI2000 Plant Canopy Analyzer device and **(c)** ground-based LAI measurements at position G1 associated with the channel's upstream cross-section

In this chapter, UAV-based images of the examined vegetated drainage channel were acquired through a MicaSense® RedEdge-M (MicaSense - Seattle, USA) multispectral camera mounted on a FIMI® Xiaomi MiDrone 4k multirotor (FIMI Robot Technology Co., Ltd. - Shenzen, China) device and supported by GPS + compass and downwelling sun sensors, as shown in figure 5.7a. A total of 1640 multispectral images were taken on average for each experimental UAV flight. After data collection in the field, the acquired images were then digitally post-processed. The radiometric calibration of the MicaSense® RedEdge-M multispectral camera's sensors (Figure 5.7b) was carried out by taking a single multispectral photograph to a reference calibration panel (Figure 5.7c) before and after each UAV flight. In detail, the radiometric calibration was performed by considering the experimental sunlight conditions in the field, based on the known panel wavelength value, as described by Mamaghani and Salvaggio [236] and Hakala et al. [237], among others. As shown in figure 5.7d, since the experimental field is in a highly urbanized area, Ground Control Points (GCP) were taken either by taking coordinates of known fixed points (building corners, street corners, and so on) or by using GCPs placed in the experimental area. All GCPs have known coordinates recorded by a differential real-time kinematics (RTK) GPS [238].

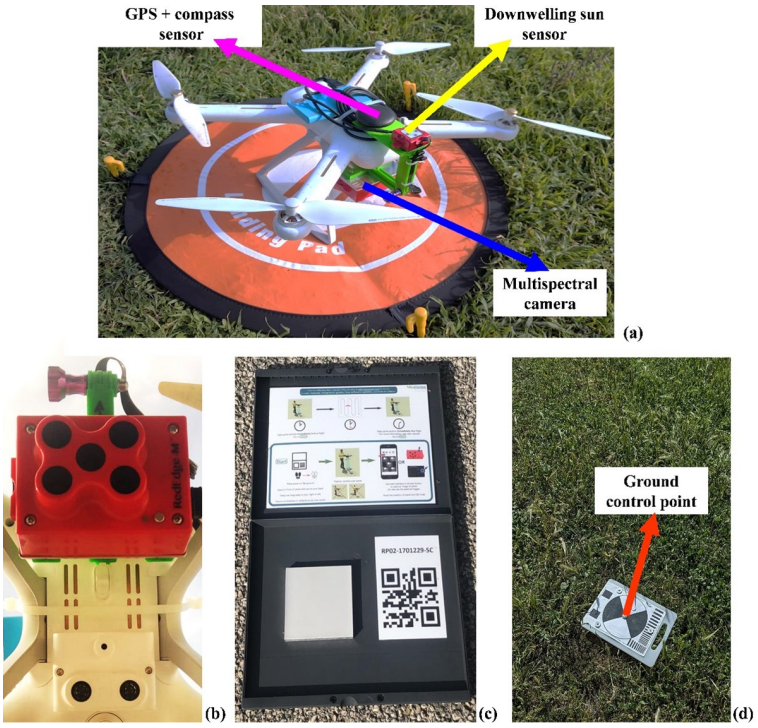


FIGURE 5.7: **(a)** View of the UAV employed in this study. Multispectral camera, downwelling sun sensor, and GPS + compass sensor are here indicated by yellow, pink, and blue arrows, respectively. **(a)** MicaSense® RedEdge-M multispectral camera, fixed as payload on the selected UAV. **(c)** Reference calibration panel employed for calibrating the multispectral camera to the field sunlight condi-tions and **(d)** Ground control point overview (orange arrow)

In this chapter, digital image and orthorectification processing were performed according to the following workflow:

1. Import photos and manually remove image outliers: photos taken before and during UAV take-off, during UAV landing, photos outside the boundaries of the experi-mental vegetated drainage channel;

2. Conversion of geotagged images (WGS84) GPS coordinates to match GCPs coordinate system;

3. Photo alignment and import GCPs list (also include the accuracy of 3D coordinates X/Y/Z);

4. Verify and link markers to images. It was feasible to mark each GCP in 3–6 images since the acquired images and markers have the same coordinate reference system;

5. Sparse point cloud cleaning, removing all points with reprojection error higher than 1, camera alignment optimization, and dense cloud building;

6. Digital Elevation Model (DEM) building from dense cloud and orthomosaic based on DEM.

MicaSense® RedEdge-M multispectral camera's most relevant parameters and their ranges are summarized in the next table 5.1.

TABLE 5.1: Main parameters of the MicaSense® RedEdge-M multispectral camera employed in this chapter, and ranges

Parameters	**Ranges**
Weight	$170g$ (Including DLS)
Dimensions	$9.4cm * 6.3cm * 4.6cm$
Power	$4.2V$ - $15.8V$, $4W$ nominal, $8W$ peak
Spectral Bands	Narrowband: Blue, Green, Red, Red Edge, Near IR
Ground Sample Distance	$8.2cm/pixel$ (per band) at $120m$ AGL
Max Capture Speed	1 capture per second (all bands), 12-bit RAW

The UAV device employed here to acquire and then digitally process the multispectral images of the 9 – $10m$ high *Arundo donax* stands covering the examined vegetated drainage channel is shown in figure 5.8a. Each experimental UAV flight plan was set to obtain the highest quality of image acquisition in terms of the camera sensors' coverage, as displayed in figure 5.8b.

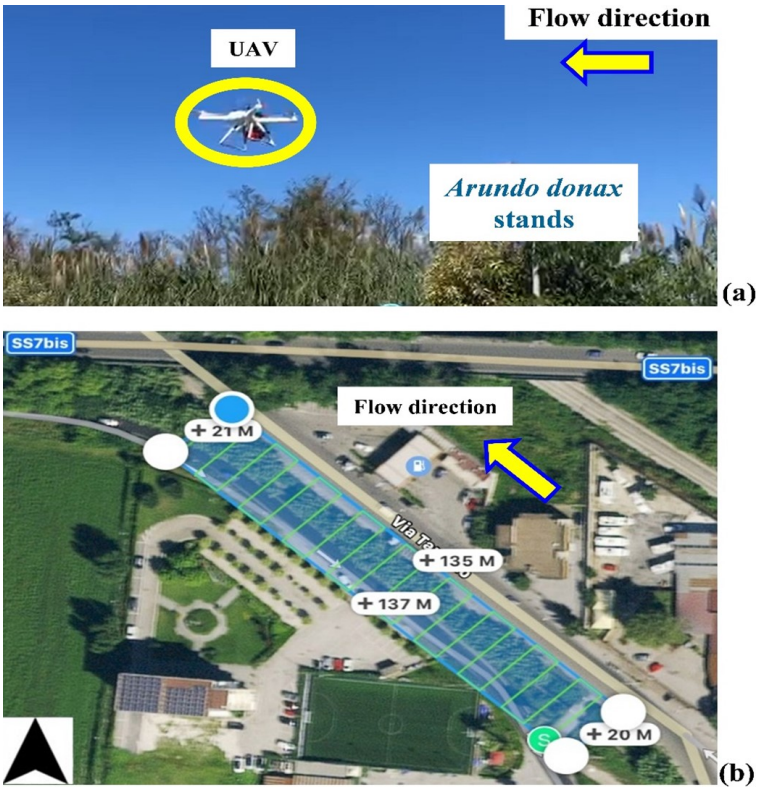


FIGURE 5.8: **(a)** UAV (yellow ellipse) and **(b)** experimental flight plan overviews. The yellow arrow indicates the flow direction

In this chapter, the UAV flight plan was set to an altitude and a flight speed equal respectively to $20m$ and $3ms^{-1}$, this way it was possible to obtain an average overlap between two consecutive multispectral photos of 75% both vertically (along

the flight direction) and horizontally (orthogonal to the flight direction) as well as minimize the influence of wind turbulence generated by UAV rotors on the examined *Arundo donax* stands. All the multispectral images acquired here were then processed via a structure-from-motion algorithm to rectify, stitch, and assemble the orthomosaic of the vegetated drainage channel. UAVs are considered as a fast way to assess riparian vegetation indicators such as riparian stands LAI in hardly accessible areas like abandoned vegetated streams. The digital processing of multispectral images acquired through UAV devices represents a useful tool to remotely observe the growth in riparian vegetation cover and, therefore, the hydrodynamic behavior of vegetated open channels colonized by riparian stands in natural phenological conditions. The first outcome of this research is represented by the direct correlation between NDVI derived by UAV-acquired multispectral images ($NDVI_{UAV}$) and LAI^* measurements of the examined riparian vegetation cover, obtained by adopting the same methodology reported in previous precision agriculture studies on UAV-based crop production indicators [239, 240]. Riparian vegetation $NDVI_{UAV}$ maps are extremely useful for flooding risk management in natural and urban vegetated areas. In detail, $NDVI_{UAV}$ estimations of the $9 - 10m$ high *Arundo donax* stands were easily derived by digitally processing UAV-acquired multispectral images by applying the following equation obtained on the basis of the spectral reflectance wavelengths ranging between $680nm$ and $785nm$ [241, 242]:

$$NDVI_{UAV} = \frac{\rho_{NIR} - \rho_{RED}}{\rho_{NIR} + \rho_{RED}} \tag{5.3}$$

where ρ_{NIR} and ρ_{RED} are the experimental spectral reflectance measurements of the examined *Arundo donax* stands acquired by the multispectral camera's sensors in the Red-Edge and Near-Infrared regions, respectively. The typical values of the five spectral bands and center wavelength acquired by the multispectral camera's sensors are shown in table 5.2, where the Band Number and corresponding names, the center of each spectral band in terms of Wavelength (nm), and the Bandwidth (nm) in terms of Full Width at Half Maximum (FWHM) are summarized.

TABLE 5.2: Spectral bands acquired by the multispectral camera's sensors and features: Band Number, Band Name, Center Wavelength (nm), and Bandwidth (nm) in terms of Full Width at Half Maximum (FWHM)

Band Number	**Band Name**	**Center Wavelength (nm)**	**Bandwidth FWHM (nm)**
1	Blue	475	20
2	Green	560	20
3	Red	668	10
4	Red Edge	717	10
5	Near IR	840	40

In this chapter, the LAI^* map associated with the $9 - 10m$ high *Arundo donax*

stands covering the examined vegetated drainage channel was obtained based on a comparison performed between $NDVI_{UAV}$ map and LAI^* measurements, both corresponding to the same 10 measuring pixels randomly distributed across each of the 30 channel's cross-sections, for a total of 300 pairs of data. $NDVI_{UAV}$ values were computed from a multispectral orthomosaic created in Agisoft® Metashape Pro v1.6 software via raster algebra operations to stitch together all UAV-acquired multispectral images were aiming at creating a single digital image of the examined vegetated drainage channel. The 300 pairs of data were randomly split into a training dataset composed of 200 pairs of data, employed to calibrate the linear regression and a test dataset composed of 100 pairs of data, to validate it, as suggested by Arsenault et al. [243], among others.

Following equation 5.2, the average LAI^* referred to each of the 30 examined channel's cross-sections and those estimated by applying the linear regression law obtained between $NDVI_{UAV}$ and LAI^* were employed to calculate the cross-sectional bulk drag coefficients ($\overline{C_D}$), respectively indicated as $\overline{C_{D_{ob.}}}$ and as $\overline{C_{D_{es.}}}$. By directly comparing them, it was possible to analyze the impact of the uncertainty in UAV-based LAI^* (LAI^*_{UAV}) values on the real-scale bulk drag coefficients ($\overline{C_D}$) predictions associated with the vegetated drainage channel examined in this study, to assess the reliability in employing them in more complex ecohydraulic numerical models of real vegetated streams fully covered by riparian weed species as Arundo donax stands at field scale.

5.1.3 Results

Flow Velocity Measurements

For the sake of brevity, figure 5.9a and figure 5.9b show the experimental cross sectional distributions of streamwise velocity components u (ms^{-1}) at respectively the only upstream and downstream cross sections of the vegetated drainage channel fully covered by $9 - 10m$ high *Arundo donax* stands.

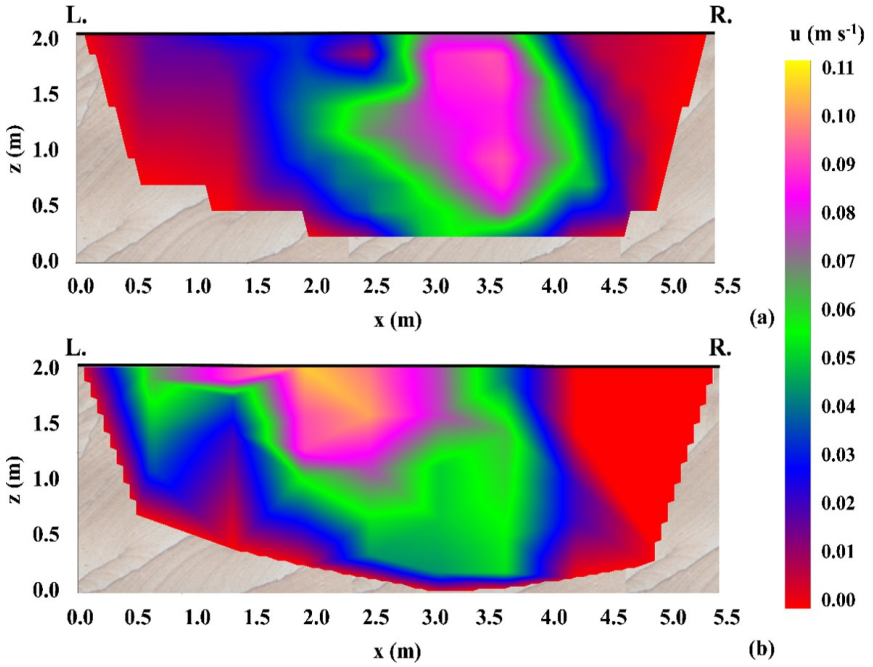


FIGURE 5.9: Experimental cross sectional distributions of streamwise velocity components u (ms^{-1}) at the vegetated drainage channel's **(a)** upstream and **(b)** downstream cross sections

In table 5.3 are summarized the experimental values of flow area A (m^2), wetted perimeter χ (m), hydraulic radius R (m), and flow average velocity U ($ms-1$) computed at the same 30 channel's cross sections selected for comparing LAI^* and $NDVI_{UAV}$.

TABLE 5.3: Values of the hydraulic parameters of the 30 measuring channels' cross sections: flow area A (m^2), wetted perimeter χ (m), hydraulic radius R (m), and flow average velocity U (ms^{-1})

Cross-Section	A **(**m^2**)**	χ **(**m**)**	R **(**m**)**	**U (**ms^{-1}**)**
1	8.695	8.120	1.071	0.01725
2	8.720	8.060	1.082	0.01720
3	8.705	8.115	1.073	0.01723
4	8.730	8.240	1.059	0.01718
5	8.656	8.145	1.063	0.01733
6	8.673	8.116	1.069	0.01730
7	8.689	8.148	1.066	0.01726
8	8.711	8.161	1.067	0.01722
9	8.654	8.162	1.060	0.01733
10	8.667	8.116	1.068	0.01731
11	8.734	8.129	1.074	0.01717
12	8.697	8.123	1.071	0.01725
13	8.710	8.144	1.069	0.01722
14	8.630	8.112	1.064	0.01738
15	8.724	8.130	1.073	0.01719
16	8.686	8.126	1.069	0.01727
17	8.724	8.115	1.075	0.01719
18	8.710	8.120	1.073	0.01722
19	8.700	8.112	1.072	0.01724
20	8.600	8.134	1.057	0.01744
21	8.670	8.171	1.061	0.01730
22	8.768	8.152	1.076	0.01711
23	8.713	8.114	1.074	0.01721
24	8.555	8.105	1.056	0.01753
25	8.668	8.130	1.066	0.01731
26	8.668	8.160	1.062	0.01731
27	8.665	8.117	1.068	0.01731
28	8.760	8.120	1.079	0.01712
29	8.631	8.128	1.062	0.01738
30	8.650	8.130	1.064	0.01734

Arundo donax Stands' $NDVI_{UAV}$ Map

As suggested by Jimenez-Berni et al. [244], it was feasible to obtain the so-called Canopy Height Model (CHM) map in figure 5.10a, representing the map of *Arundo donax* stands' heights directly from DEM based on the digital processing of the UAV-acquired multispectral images. Then, the $NDVI_{UAV}$ map was retrieved by applying

equation 5.3 at each pixel of the UAV-acquired multispectral images, as shown in figure 5.10b.

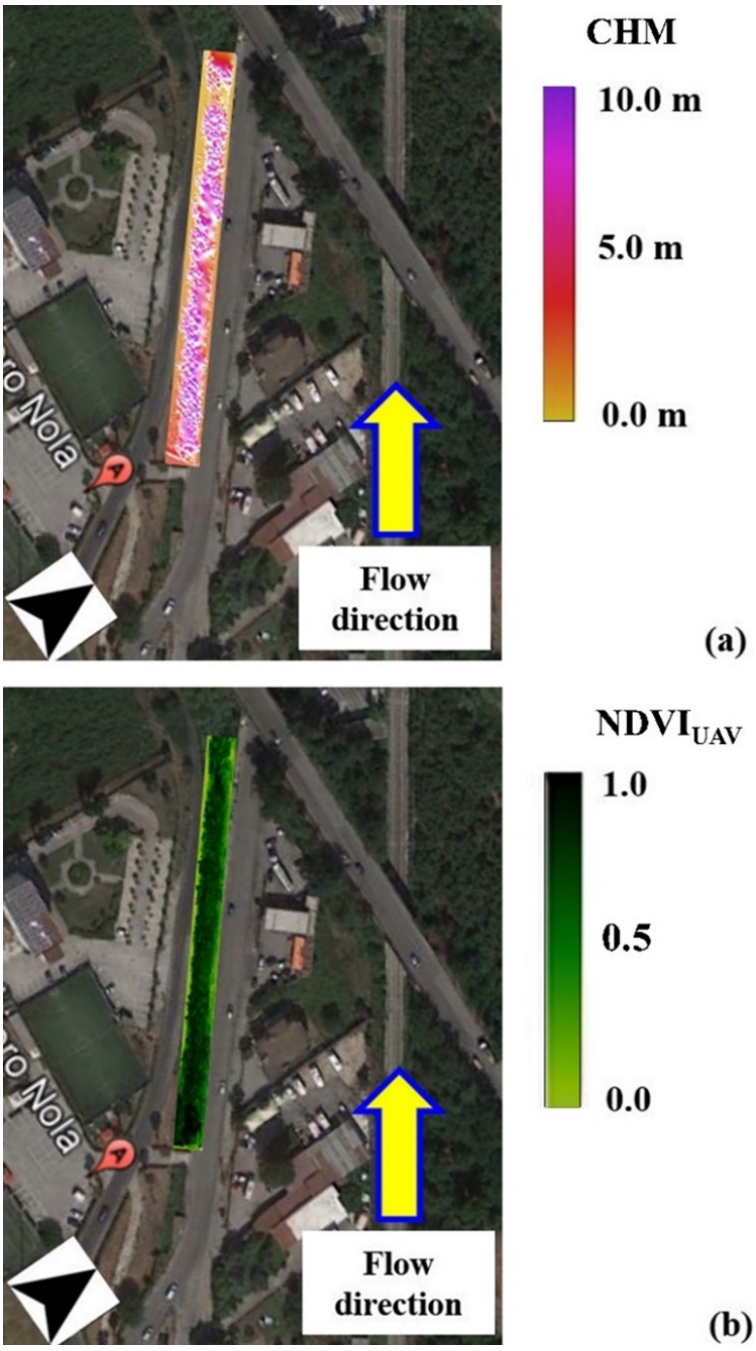


FIGURE 5.10: UAV-based **(a)** Canopy Height Model (CHM) and **(b)** Normalized Difference Vegetation Index ($NDVI_{UAV}$) maps of the 9 – 10m high *Arundo donax* stands. The yellow arrow indicates the flow direction

It is important to highlight here that UAV-acquired multispectral images refer to the reflectance properties of the upper portions of *Arundo donax* stands' canopy and stems, which are necessarily correlated to the lower stands' portion effectively interacting with water flow dynamics, due to phenological and morphological processes,

inevitably affected by the whole *Arundo donax* growth [245]. A complete analysis of the interplay between all the biological components of green volumes involved in these processes is out from the purposes of this chapter and will be examined more in detail in further works on riparian *Arundo donax* and *Phragmites australis* stands' growth. Consequently, the further comparative analysis can be considered as properly representative of the real phenological evolution of the examined riparian vegetation species.

$NDVI_{UAV}$ and LAI^* Correlation, Calibration and Validation

As reported in figure 5.11, in order to detect the degree of dependence between LAI^* and $NDVI_{UAV}$ and to establish a possible correlation between the two experimental vegetation indices corresponding to the examined $9 - 10m$ high *Arundo donax* stands, a linear regression analysis was performed on 300 pairs of data recorded at 30 channel's cross-sections, based on split sampling validation method. In detail, 200 data were analyzed for the training set to calibrate the regression linear law, and the remaining 100 data were employed for validating it.

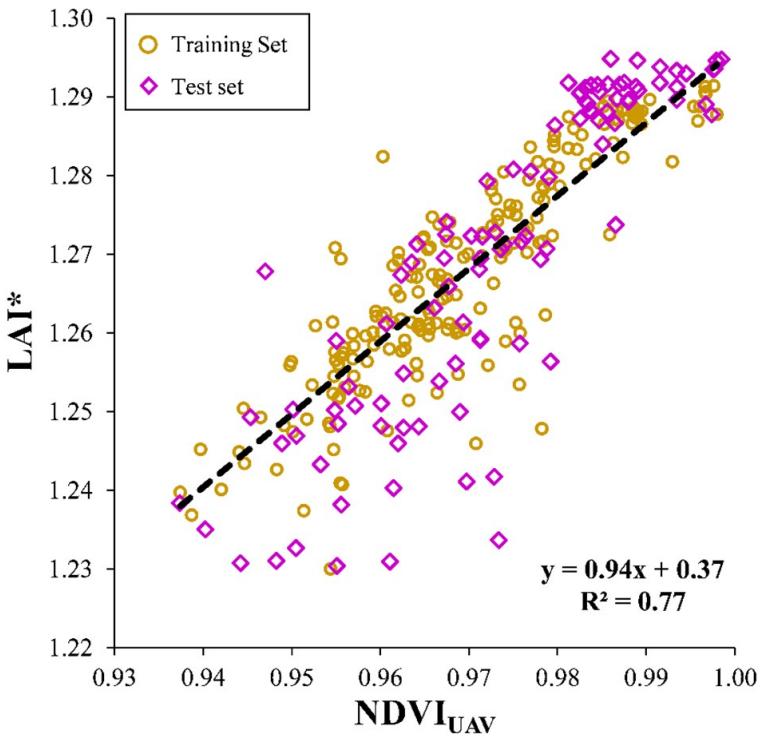


FIGURE 5.11: Linear law between $NDVI_{UAV}$ and LAI^* referred to the 30 channel's cross-sections, for a total of 300 pairs of data. The Training (200 pairs of data) and Test datasets (100 pairs of data) are indicated here by yellow unfilled circles and purple unfilled diamonds, respectively

A high correlation existing between $NDVI_{UAV}$ values and LAI^* ones was observed in figure 5.11, testified by a coefficient of determination R^2 equal to 0.77. This demonstrates that $NDVI_{UAV}$ represents a good predictor for LAI^* measurements

in vegetated flows fully covered by senescent *Arundo donax* stands. The regression model obtained here is a linear function with an angular coefficient equal to 0.94 and a low intercept equal to 0.37. Based on a test set composed of 100 pairs of data, the validation of the linear regression confirmed the high level of correlation existing between the two riparian vegetation indices analyzed here ($R^2 = 0.75$). The LAI^* values obtained by applying the linear law displayed in figure 5.11 to all the $NDVI_{UAV}$ map pixels are indicated as LAI^*_{UAV}, and the corresponding map is presented in figure 5.12 to visualize LAI^*_{UAV} distribution along the whole vegetated drainage channel fully covered by 9 – 10m high *Arundo donax* stands.

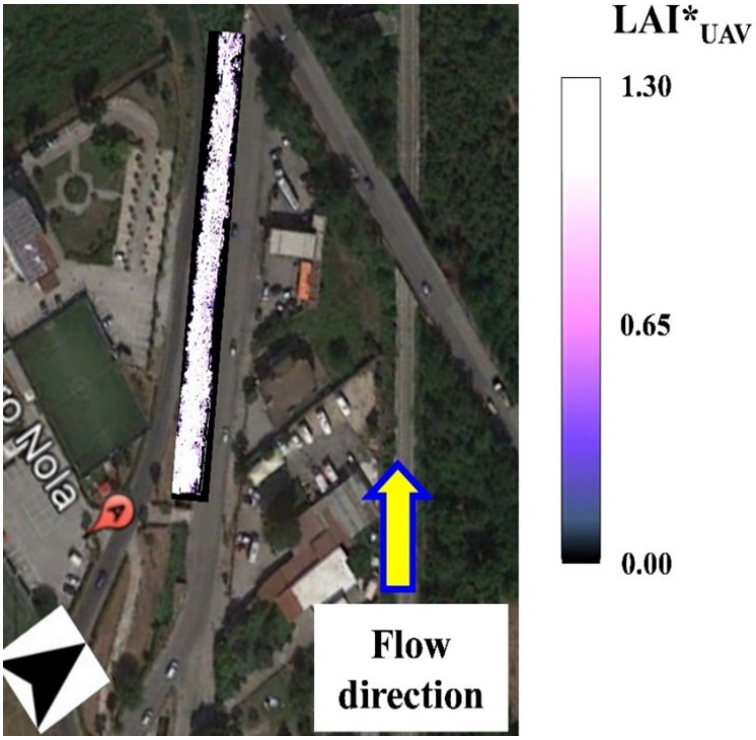


FIGURE 5.12: LAI^*_{UAV} map of the vegetated drainage channel fully covered by 9 – 10m high *Arundo donax* stands. The yellow arrow indicates the flow direction

Comparison of Bulk Drag Coefficients ($\overline{C_D}$) Predictions

Figure 5.13 reports a comparative analysis performed between bulk drag coefficients ($\overline{C_D}$) predictions obtained by employing respectively ground-based LAI^* ($\overline{C_{D_{ob.}}}$) and LAI^*_{UAV} estimations ($\overline{C_{D_{es.}}}$) of the examined Arundo donax stands. A linear regression analysis was carried out to easily evaluate the impact of the performance of UAV-based multispectral imagery on ($\overline{C_D}$) predictions accuracy, based on the vegetative flow resistance model proposed by Västilä and Järvelä for rigid and emergent riparian stands [220].

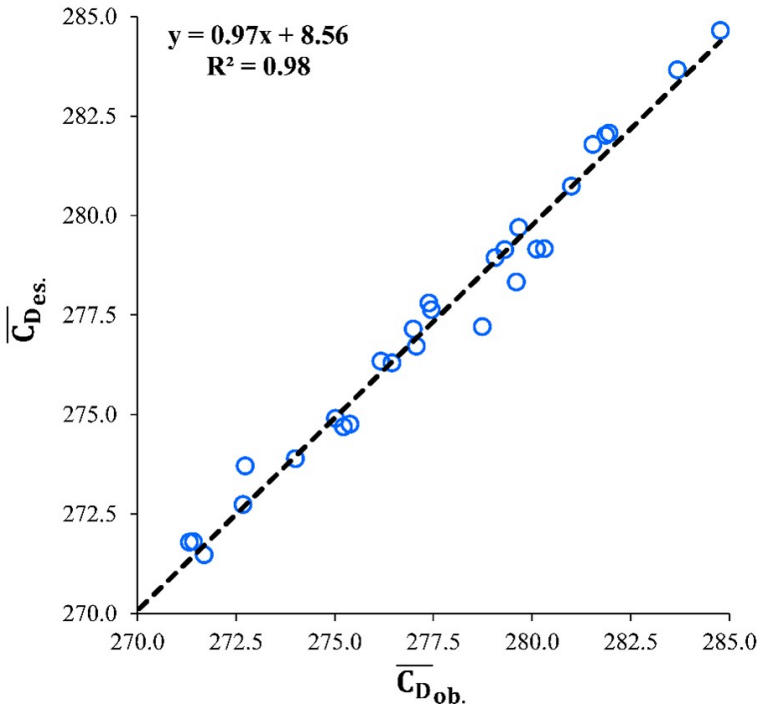


FIGURE 5.13: Comparison of bulk drag coefficients obtained through average LAI^* and LAI^*_{UAV} estimations at the 30 channel's cross sections, respectively indicated as ($\overline{C_{D_{ob.}}}$) and ($\overline{C_{D_{es.}}}$)

It emerges from the comparative analysis shown in figure 5.13 that $\overline{C_{D_{ob.}}}$) and $\overline{C_{D_{es.}}}$) assume values extremely comparable, almost identical, as indicated by a very high coefficient of determination $R^2 = 0.98$.

5.1.4 Discussion

UAV-based remote sensing allows recording riparian vegetation dimensional traits in areas whose access is difficult. UAVs are widely used in precision agriculture and forestry studies and applications to calculate and then mapping riparian vegetation indexes. Their use can be extended to the prediction LAI in areas where it is not possible to easily harvest riparian vegetation samples, such as abandoned vegetated open channels. The outcomes of the study in this chapter showed a good correlation between ground-based LAI measurements and NDVI derived by digital processing of UAV-acquired multispectral images. The outcomes in terms of LAI distributions agree with the main findings of the study carried out by Fagnano et al. [246] who experimentally recorded LAI of *Arundo donax* stands during their whole phenological evolution for one year. The accuracy of LAI estimations of the portion of $9 - 10m$ high Arundo donax stands effectively interacting with water flow, derived from the $NDVI_{UAV}$ map obtained from UAV-acquired multispectral image is highly comparable with the outcomes of the study proposed by Tan et al. [247], which analyzed different types of wheat plants at distinct growth and foliation stages. Indeed, they obtained values of coefficients of determination R^2 equal to 0.78, 0.77, and 0.76 for respectively erectophile, middle, and planophile samples, equal to that observed here

($R^2 = 0.77$), corresponding to a phenological condition of *Arundo donax* senescence. The main assumption made here in terms of ground-based LAI estimations leads to the morphological and phenological correlations existing between the upper riparian canopy and lower stems. This issue can be overcome in future studies by accurately monitoring the 3D morphometry of *Arundo donax* stands' covers at field scale in many different vegetated streams in natural conditions. It was observed that bulk drag coefficients predictions are independent, or very little dependent, on the uncertainty in LAI^* derived by digital processing of UAV-acquired multispectral images of *Arundo donax* stands covering the examined vegetated drainage channel, indicating that UAV-based NDVI is a suitable parameter to be employed in the prediction of bulk drag coefficients of senescent riparian *Arundo donax* stands. It is extremely interesting since, as reported in the studies performed by Luhar and Nepf [248] and Zhang and Nepf [249], an accurate prediction of the actual bulk drag coefficients is crucial for analyzing the effects of the combination of stems reconfiguration and plants canopy sheltering on the mean and turbulent hydrodynamic behavior of real vegetated water bodies. Differently from the study case in this chapter characterized by the massive presence of invasive and senescent *Arundo donax* stands, Zhang and Nepf [249] have examined experimental data retrieved under a condition of patchy riparian vegetation, which inevitably limited the reduction of bulk drag coefficients ($\overline{C_D}$) compared to the experimental field condition of total riparian vegetation abandonment observed in our real-scale analysis.

5.1.5 Conclusions

Bulk drag coefficients ($\overline{C_D}$) associated with a real vegetated drainage channel fully covered by 9 – 10m high *Arundo donax* stands were predicted and validated here, based on the LAI map derived by NDVI assessments obtained through digital processing of UAV-acquired multispectral images of the riparian vegetation cover.
$NDVI_{UAV}$ map values were correlated to ground-based LAI measurements of *Arundo donax* portion effectively interacting with flow to obtain LAI^* maps to estimate the bulk drag coefficients ($\overline{C_D}$) associated with 30 cross-sections identified along the examined vegetated drainage channel. It was possible applying the predictive model of vegetative flow resistance proposed by Västilä and Järvelä [220] once the experimental values of the main hydraulic and hydrodynamic features measured at the same channel's cross-sections were known. The strong capability of UAV-acquired multispectral imagery in estimating the bulk drag coefficients ($\overline{C_D}$) of the examined vegetated drainage channel fully covered by 9 – 10m high*Arundo donax* stands was then demonstrated by a comparative analysis performed between ($\overline{C_D}$) prediction assessed by considering the average observed and UAV-based LAI measurements at the same 30 channel's cross-sections, respectively. Also, it was demonstrated here that NDVI data recordings based on UAV-acquired multispectral images can be exploited to develop further methods for predicting actual LAI^* or other riparian and

aquatic vegetation indices, such as those based on deep learning/machine learning algorithms, already widely validated in many precision agriculture and rainfall prediction studies. Further studies are certainly undergoing to develop these algorithms and making even faster the assessment of the most relevant changes in water flow dynamic features of vegetated open channels associated with different riparian vegetation species, under many distinct ecohydraulic conditions, by also analyzing the key bio-mechanical and morphometric properties of riparian stands at micro-scale. The results of the study in this work represent a satisfactory advance in the monitoring of riparian vegetation dimensional properties in the field, particularly in real-scale models and simulations of vegetated flows to be employed as supporting tool for the proposal of the most appropriate mitigating scenarios of flooding events in natural and urban areas surrounding vegetated water bodies.

5.2 Impact of Riparian Plants Biomass Assessed by UAV Acquired Multispectral Images on the Hydrodynamics of Vegetated Streams

5.2.1 Introduction

Ecohydraulics is a discipline based on ecology and environmental hydraulics, mixed together in order to furnish a useful tool to predict the behaviour of vegetated open channels to land and reclamation area managers and forestry engineers. The processing of Unnamed Aerial Vehicle (UAV) acquisitions is a fast way to assess riparian vegetation biomass, synthetically expressible by means of Leaf Area Index (LAI), in hard-accessible areas such as vegetated open channels in natural conditions. The work in this subchapter has the aim to integrate UAV-based plant indexes and hydraulics measurement to assess and predict how biomass change can affect the behaviour of vegetated water bodies by considering a vegetative flow resistance model based on Leaf Area Index (LAI). Defining the most appropriate management practice of riparian vegetation inside both natural and manmade water bodies is crucial for assuring a balance between a satisfactory level of hydraulic conveyance and a high environmental value of water. In the study in this subchapter, It has been analyzed the accuracy of modeled hydraulic features of vegetated open channel estimated by employing two methods of biomass assessments: ground-based LAI and UAV-derived vegetation indexes. It was possible obtaining LAI maps quickly related to Normalized Difference Vegetation Index (NDVI) maps by processing UAV-acquired multispectral images. Six values of flow average velocities, representative of the typical values of flow average velocity regimes occurring in vegetated drainage channels, were considered: 0.030, 0.035, 0.040, 0.045, 0.050, and 0.060 ms^{-1}. Thus, by applying the LAI-based flow resistance model, it was possible to compute the corresponding values of bulk drag coefficients.

5.2.2 Materials and Methods

As depicted in figure 5.14, the experimental study area is a vegetated drainage channel fully covered by emergent invasive riparian plants, located in the Municipality of Napoli (Southern Italy).

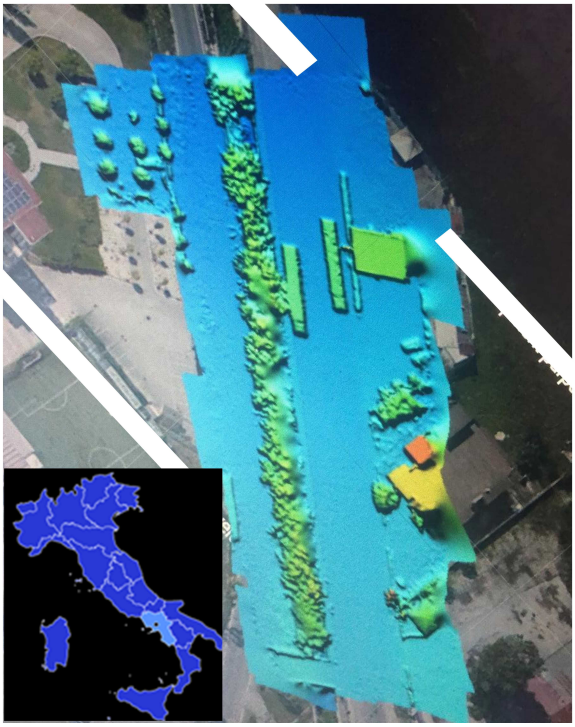


FIGURE 5.14: Experimental area overview: vegetated drainage channel in Napoli (Southern Italy). The flow direction goes from the bottom to the top of the figure

Riparian vegetation UAV-based NDVI and ground-based LAI measurements were carried out and then compared aiming at analytically predicting the hydrodynamic behaviour of the examined vegetated channel (see 5.1). As displayed in figure 5.15, the flight scheme employed in this subchapter work aimed at covering the experimental borders of the examined vegetated water body as much as possible.

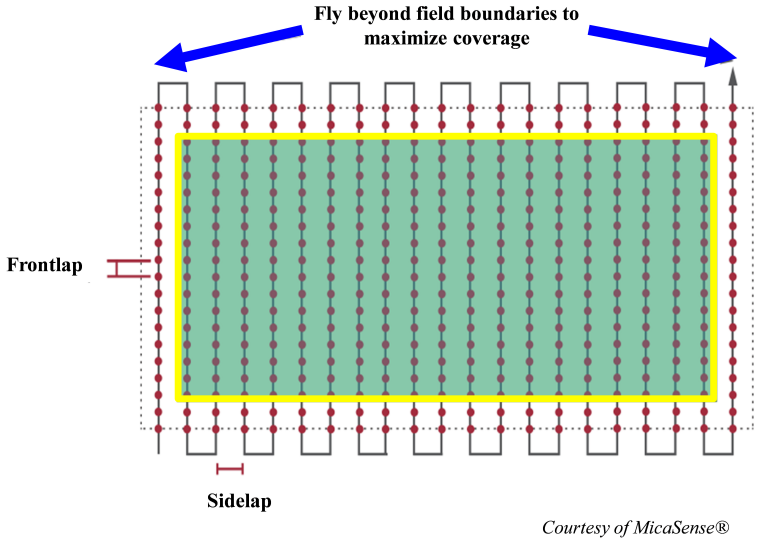


FIGURE 5.15: Experimental scheme of the flight plan selected for the present study case

5.2.3 Result and Discussion

The UAV-based and ground-based LAI measurements of the examined vegetated water body were then compared in terms of average values, corresponding to 30 random pixels located along the channel. The next figures 5.16a and 5.16b show the values of average NDVI derived from the digital processing of UAV-acquired images and ground-based LAI measurements, hereinafter respectively indicated as LAI_{G_m} and $NDVI_{UAV_m}$.

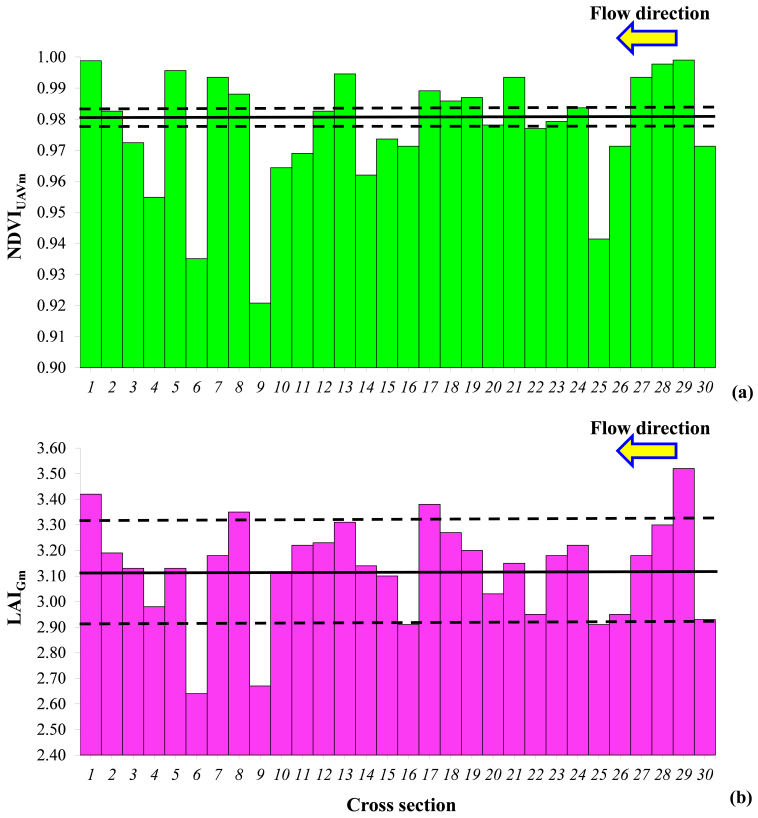


FIGURE 5.16: **(a)** $NDVI_{UAV_m}$ (green bars) and **(a)** LAI_{G_m} (magenta bars) values. The black continuous horizontal line represents the average value, while the black dashed horizontal lines represent the mean value $\pm$ standard deviation. The yellow arrow indicates the flow direction

As illustrated in figure 5.16a, $NDVI_{UAV_m}$ data are characterized by a mean value and a standard deviation respectively equal to 0.98 and 0.02. From the analysis of figure 5.16b, it can be observed that LAI_{G_m} mean value is equal to 3.13, while the standard deviation is 0.20, exactly ten times the one referred to $NDVI_{UAV_m}$. The same values appearing in figure 5.17a and figure 5.17b are represented in figure 5.17 to establish a possible correlation between the two variables.

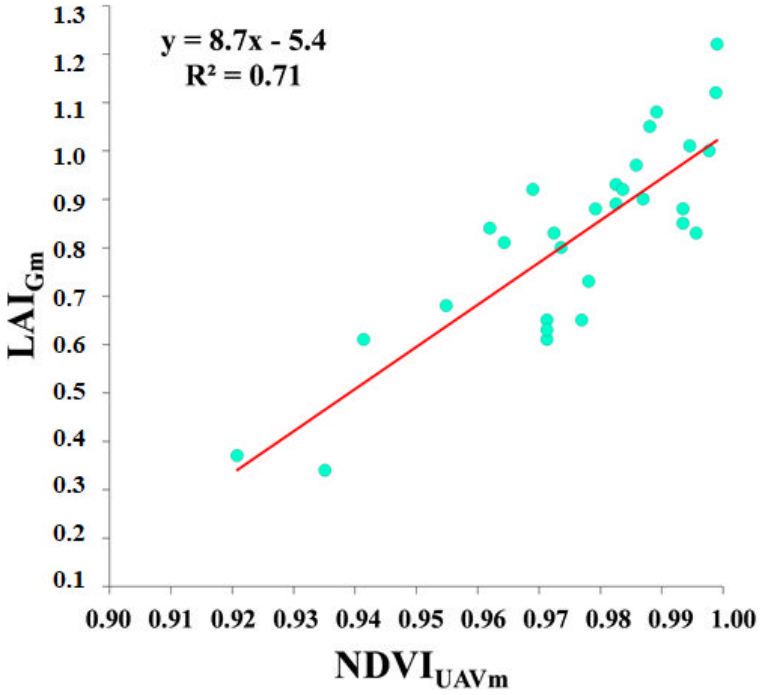


FIGURE 5.17: Linear law between $NDVI_{UAV_m}$ and LAI_{G_m}

As figure 5.17 suggests, a satisfactory correlation exists between $NDVI_{UAV_m}$ values – one for each investigated point – and LAI_{G_m} ones - again, one for each investigated location. This demonstrates that $NDVI_{UAV_m}$ can be considered a good predictor for ground-based LAI measurements in vegetated flows. The regression model here obtained is a linear function with an angular coefficient equal to 8.7 and an intercept of -5.4 with a coefficient of determination equal to 0.71. It is important to highlight here that since $LAI = -5.4$ for $NDVI = 0$ has no physical significance, the accuracy of observed linear law depends on the experimental ranges of data collected experimentally (5.1). In the next figure 5.18 are shown the average values of LAI_{G_e}, representing the LAI_G estimated under the correlation law indicated in figure 5.18.

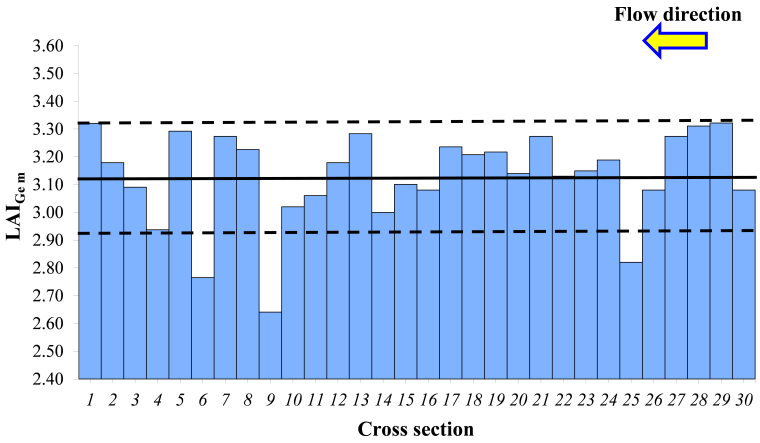


FIGURE 5.18: LAI_{G_e} (soft blue). The black continuous horizontal line represents the average value, while the black dashed horizontal lines represent the mean value $\pm$ standard deviation. The yellow arrow indicates the flow direction

To better analyze and then discuss the outcomes in terms of bulk drag coefficients variability among the 30 random pixels analyzed here, the variation ranges are represented in figure 5.19 as a function of the six increasing values of flow average velocities selected for the analyses conducted in the present study: 0.030, 0.035, 0.040, 0.045, 0.050, and 0.060 ms^{-1}.

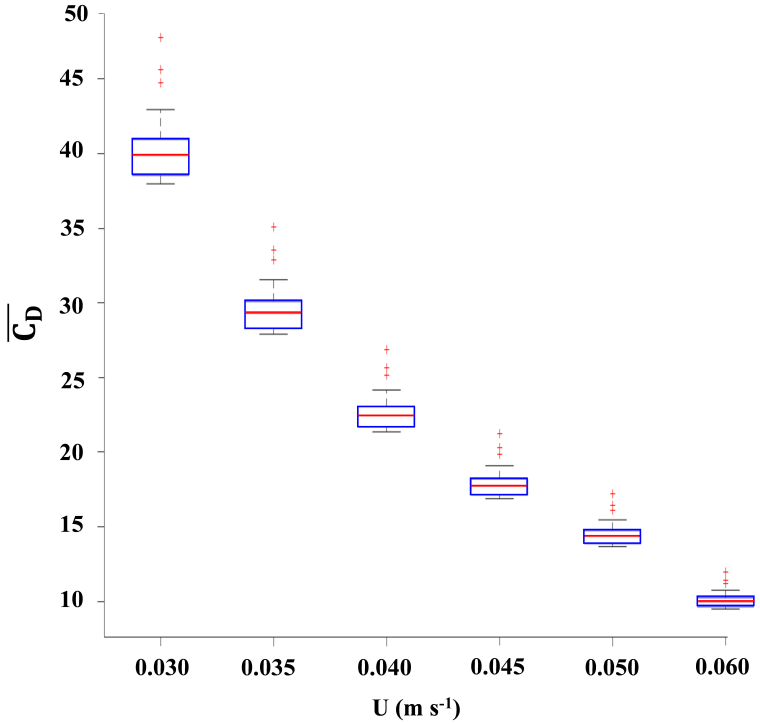


FIGURE 5.19: Box plots indicating the variabilities in drag coefficients for the six values of flow average velocities considered in the present study: 0.030, 0.035, 0.040, 0.045, 0.050, and 0.060 ms^{-1}

It emerges from the analysis of figure 5.19 that the variability in bulk drag coefficients values estimated at to the examined 30 pixels strongly decreased moving from

$U = 0.030ms^{-1}$ to $U = 0.060ms^{-1}$, testified by two corresponding standard deviations equal to 2.23 and 0.58, respectively. The number of extreme data, here defined as the bulk drag coefficients values greater than 3 times the interquartile lengths, was the same for all the data (indicated by the red crosses in figure 5.19), and equal to 3.

5.2.4 Conclusions

Starting from the comparison between LAI derived from UAV and ground-based LAI measurements based on optical devices, it was possible evaluating the impact of the accuracy of UAV techniques on the prediction of vegetated channels' fluid dynamics main features by employing predictive models based on LAI (5.1). From the preliminary results obtained in this subsection study, it was possible assessing that UAV-based biomass estimations can be effectively used in the modeling of flow dynamics in real vegetated streamlines. The promising results of this subsection study can be used in more sophisticated and complete experimental and modeling analysis on the real-scale hydrodynamics of vegetated water bodies.

5.3 Assessing The Role Of Gap Fraction On The Leaf Area Index (LAI) Estimations Of Riparian Vegetation Based On Fisheye Lenses

5.3.1 Introduction

The detection and modeling of morphometric and bio-mechanical properties of real aquatic and riparian vegetation is one of the key issues in Ecohydrology and Ecohydraulics [250]. Among many parameters indicating the status of riparian vegetation biomass, the well-known Leaf Area Index (LAI) must be considered as the most representative when dealing with vegetated rivers or water bodies in general. LAI has been measured in many different agricultural and forestry scenarios [251] but a very few studies can be addressed to this topic in the context of riparian ecosystems (see 5.1). In this subsection study, the two most important indirect methods of LAI were compared to assess the influence of gap fraction on LAI estimations of weed riparian plats: LI-COR® LAI-2000 Plant Canopy Analyzer device and Fisheye lenses. In detail, a statistical analysis of gap fraction was performed here to assess its influence on Fisheye lenses for weed riparian plants' species, covering both natural and manmade water bodies [252].

5.3.2 Materials and Methods

The Study area is represented by a vegetated drainage channel heavily colonized by woody emergent invasive riparian vegetation, located in the territory of the Municipality of Napoli (Southern Italy), as shown in figure 5.20.

FIGURE 5.20: View of the study area: vegetated drainage channel in Napoli (Southern Italy)

Twenty riparian plants were selected along the drainage channel to evaluate the sensitivity of Fisheye-based LAI to gap fraction values. The reference Leaf Area Index (LAI) values were obtained by employing the plants' transmittance-device LI-COR® LAI-2000 Plant Canopy Analyzer, as suggested by many previous ecohydrology and ecohydraulics studies and reviews (see 5.1). In figure 5.21 is shown an example of LAI-2000 Plant Canopy Analyzer measurements when dealing with weed plants.

FIGURE 5.21: Indirect Leaf Area Index (LAI) estimations by LI-COR® LAI2000 Plant Canopy Analyzer

LAI values obtained through LI-COR® device were then comparted with those obtained by gap fraction inversion by employing a system of fisheye + camera lens.

Fisheye lenses used in this subsection study are adopted since they can be easily transported through vegetated streams, and then adopted according to many experimental set-up.

5.3.3 Results

The following figure 5.22 shows the outcomes of the binarization process employed here for the Fisheye-based HD photos of the examined woody riparian plants. Each circle in figure 5.22 refers to a different gap fraction depending on a specific Digital Number (DN) threshold: 250, 200, 150, 100.

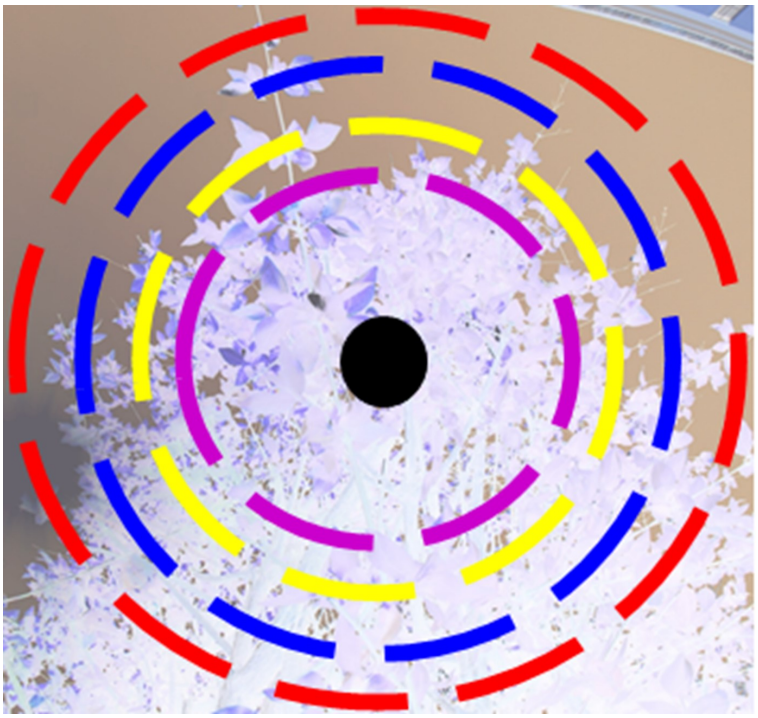


FIGURE 5.22: Example of DHP binarized associated with woody riparian plants for different gap fraction depending on DN threshold: $DN = 250$ (red dashed circle), $DN = 200$ (blue dashed circle), $DN = 150$ (yellow dashed circle), and $DN = 100$ (violet dashed circle). The black filled circle indicates the lens center

In the following figure 5.23 is reported the comparative analysis carried out between LAI values computed by employing LI-COR® LAI-2000 Plant Canopy Analyzer device and Fisheye system associated with as binarization threshold at $DN = 100$, respectively indicated as LAI_{GAP1} and LAI_{GAP2}.

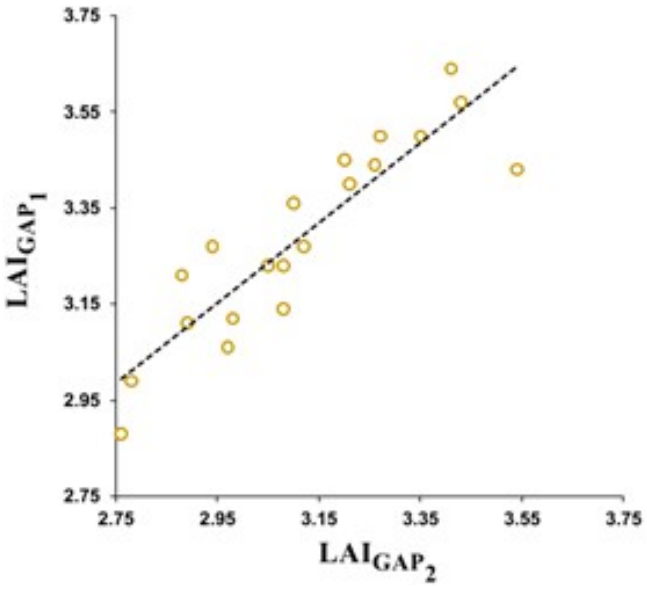


FIGURE 5.23: Comparative analysis between LAI_{GAP1} and LAI_{GAP2} values for weed riparian species

As it can be easily detected in figure 5.23, there is a good correlation between LAI_{GAP1} and LAI_{GAP2} associated with weed riparian plants, as shown by a coefficient of determination $R^2 = 0.76$.

5.3.4 Discussion

The results obtained in the present study agree with previous studies on the influence of gap fraction on the Leaf Area Index (LAI) of emergent riparian vegetation plants. In this subsection work, the effect of gap fraction was investigated by considering different threshold values of Digital Number (DN), growing from 100 to 250, in order to identify the optimal gap fraction value associate to DN, to which corresponded the highest accuracy in terms of Leaf Area Index (LAI) prediction derived by LI-COR® LAI-2000 Plant Canopy Analyzer device. The good correlation observed between LAI derived by the two methods employed here testified that gap fraction can be also analyzed by simpler techniques as commercial camera and fisheye lenses. The very promising results presented here can be effectively considered as primary insight in terms of proposal of new predictive models of the main hydrodynamic properties of vegetated channels, as indicated in the following figure 5.24, to be then employed for the calibration and validation of numerical hydrodynamic models and simulations.

FIGURE 5.24: Example of experimental ecohydraulic measurements in terms of streamwise velocity components, to be employed for calibrating and validating Computational Fluid Dynamics (CFD) simulations of real vegetated flows

In detail, the riparian vegetation elements can be better modeled and simulated from the bio-hydrodynamic perspective by considering their actual morphometric and phenological characteristics as the Leaf Area Index (LAI) proposed here. In the following figure 5.25 is reported and example of CFD simulation of a real vegetated flow colonized by circular weed riparian plants.

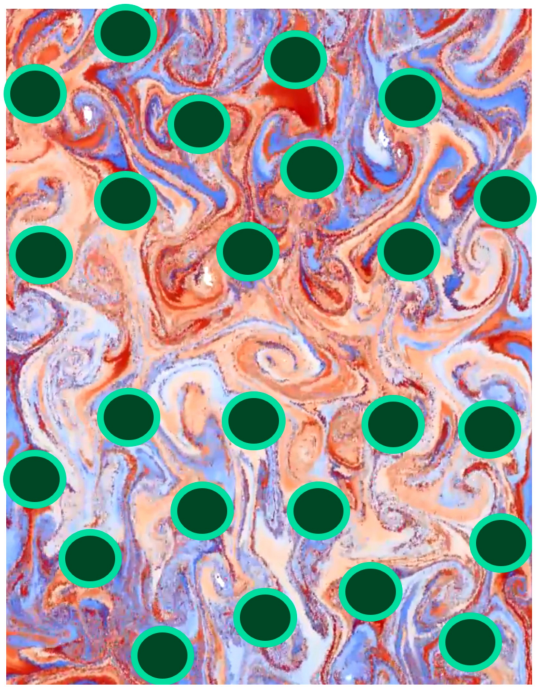


FIGURE 5.25: Example of Computation Fluid Dynamic (CFD) simulation snapshot for a real vegetated open channel. The flow direction goes from the bottom to the top of the figure

As depicted in figure 5.25, the direct analysis of the hydrodynamic behaviour of real vegetated steams can furnish highly detailed trends and patterns of vegetation-induced turbulence and vortices or eddies.

5.3.5 Conclusions

The work presented in this subsection well demonstrates that gap fraction differently influences LAI of riparian vegetation based on the imagery of Fisheye Lenses, depending on the distinct species and phonological stages analyzed in this study. In fact, it was observed that the phenological evolution and the biomechanical behaviour of riparian vegetation stems and leaves sensibly affected the performance of the binarization of images acquired by Fisheye lenses. The chance of utilizing the main findings obtained in this study represent an interesting advance in the characterization of real riparian stands covering both manmade and natural vegetated water bodies, aiming at properly assessing their actual hydrodynamic features.

5.4 Field-scale remote sensing of ecohydraulic and ecohydrologic phenomena in vegetated waterways

5.4.1 Introduction

Plants embody the physical borders between vegetated water bodies and surrounding ground, with important effects on natural habitats growing within them, as

shown in figure 5.26.

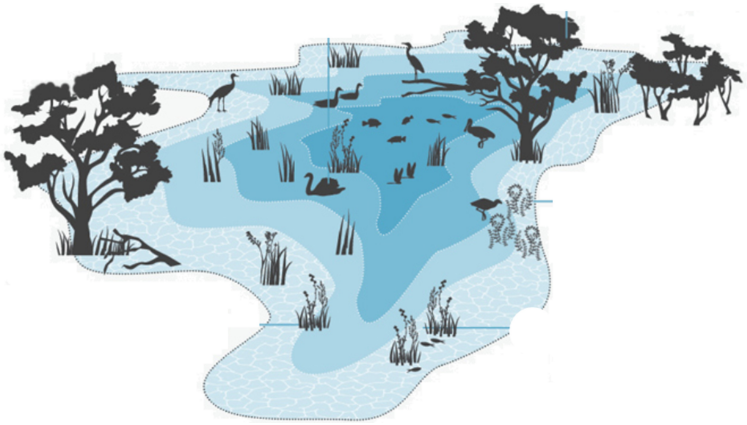


FIGURE 5.26: Scheme of terrestrial and aquatic ecosystems developing in vegetated wetlands and lowlands

The main eco-hydrodynamic features of both manmade and natural vegetated channels are influenced by riparian plants' morphometry and architecture, strongly affecting their structural and bio-mechanical properties. The values of these real-scale properties are obtainable from other physically-based vegetation indices as Leaf Area Index (LAI), Normalized Difference Vegetation Index (NDVI), and Plant Phenology Index (PPI) (see 5.1). It was demonstrated that, among other indices, LAI represents a highly reliable predictor of plants' canopy and distribution covering vegetated waterways (5.4.4). Figure 5.27 reports the ecohydraulic classification of riparian vegetation growing within vegetated watercourses: emergent, floating, and submerged, occurring when stems and canopy develop above, on, and below the water table, respectively.

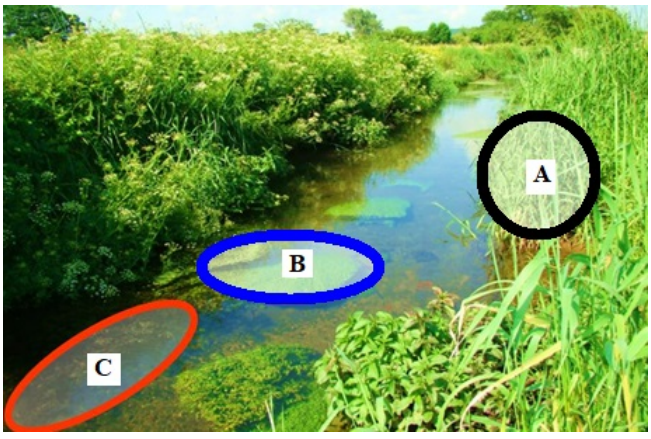


FIGURE 5.27: Detailed overview of field-scale ecohydraulic riverine and acquatic vegetation classification: **A)** emergent plants (black circle), **B)** floating plants (blue ellipse), and **C)** submerged plants (orange ellipse)

As demonstrated in many notable ecohydraulic studies and reviews, LAI is a

crucial element in the prediction of the hydraulic roughness of vegetated flows covered by rigid plants, given the linear association existing between LAI and the so-defined vegetative global flow resistance (5.1). Among other remote sensing techniques, drone-based imagery can be considered as highly suitable for characterizing the full-scale eco-hydrodynamic behaviour of vegetated waterways derived by LAI trends, since it can be simply employed in almost all environmental and orographic contexts in the field (4.1). The recent technological improvements on real-scale scans based on portable gaming-type devices are considered by environmental, forestry, and hydraulic engineers as a useful tool for the indirect estimation of rigid plants and stands' LAI. In this subsection study, the accuracy of this method in predicting LAI values associated with rigid riparian plants was compared to those derived by multispectral images analysis retrieved by drone-based remote sensing (4.1) representing a very important starting point in the calibration of vegetative flow resistance formulas for predicting the main ecohydrologic and ecohydraulic features of vegetated water systems colonized by rigid aquatic and riverine plants and patches.

5.4.2 Materials and Methods

Thirty rigid plants covering an Italian vegetated ditch were monitored in the present study case to compute LAI through real-scale scan derived by a commercial gaming-type portable device and drone-based HD images acquired through an embedded multispectral camera. Figure 5.28 illustrates an example of the commercial-type drone employed here. After a preliminary calibration stage of the multispectral camera to the environmental radiative sunlight conditions, the riparian plants' stems and canopies were then monitored through drone-based multispectral acquisitions to obtain LAI via raster-algebra algorithms.

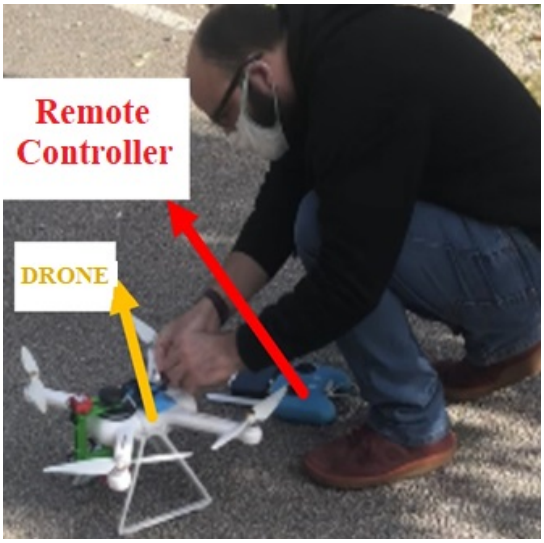


FIGURE 5.28: Detailed overview of the commercial-type drone employed in the present study case: the yellow and the red arrows indicate the drone equipped with a multispectral camera and its remote controller, respectively

Figure 5.29a and figure 5.29b display frontal and above real-scale scans of the examined plants, respectively. Thus, each scan was processed by using free imagery software for calculating LAI of the thirty rigid plants examined in the present study case.

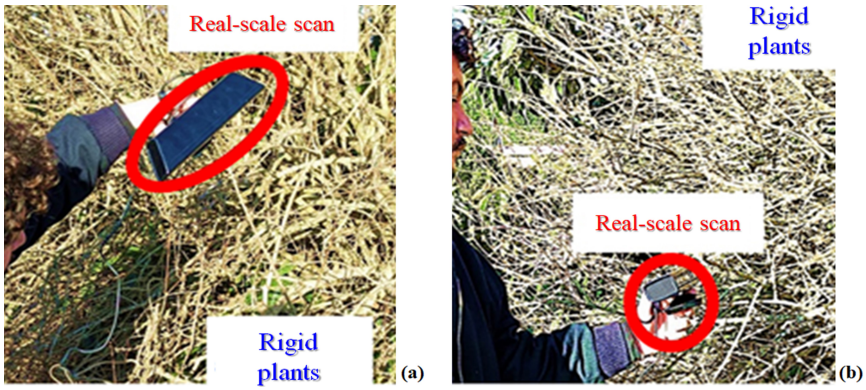


FIGURE 5.29: Real-scale scans of rigid plants colonizing an Italian vegetated ditch: **(a)** above and **(b)** frontal scans. The red ellipse indicates the gaming-type device

Figure 5.30 shows the results of the real-scale scans for one of the rigid plants analyzed here.

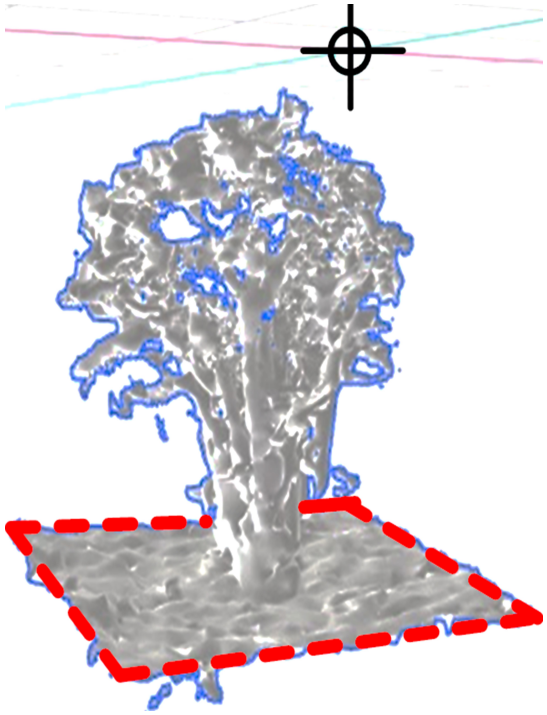


FIGURE 5.30: Digital processing of real-scale scan for a single rigid plant, with ground level indicated by red dashed lines

First, LAI values of the thirty rigid plants retrieved by processing the real-scale scans (hereinafter indicated as LAI_{ω}) were evaluated and then compared to those computed by multispectral UAV-based orthomosaic analysis (hereinafter indicated

as LAI_δ) to identify a convenient relation between the two parameters analyzed here. LAI_{DRONE} values were computed by following the indications of Lama et al. [253], on the basis of experimental correlations retrieved between NDVI and ground-based LAI, as shown in figure 5.31.

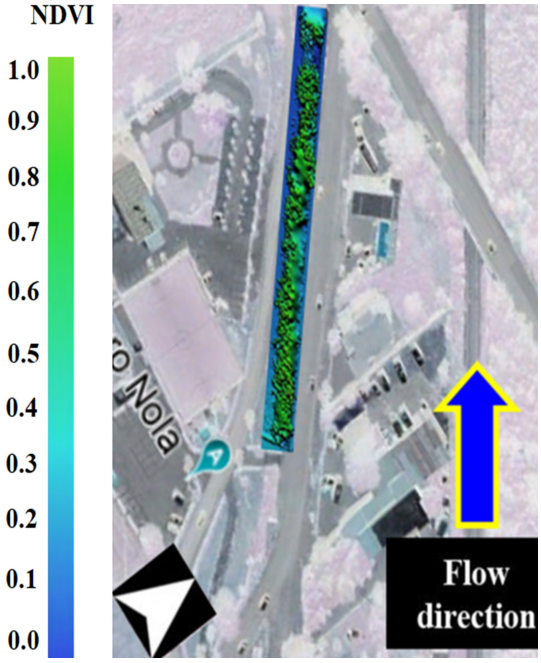


FIGURE 5.31: Drone-based NDVI for a fully vegetated abandoned ditch

In detail, the NDVI othomosaic map was obtained by combining several drone-based multispectral orthophotos, which are geometrically corrected (orthorectified) images, to achieve a uniform scale in output with given projection function over the entire map.

5.4.3 Result and Discussion

Table 5.4 reports the values of LAI_ω and LAI_δ derived in this study for the thirty examined rigid plants.

TABLE 5.4: LAI_ω and LAI_δ values computed for the rigid plants surveyed in this subsection experimental study case

LAI_ω	LAI_δ
3.20	3.14
3.07	3.12
2.50	3.00
2.76	2.81
3.10	3.21
3.21	3.25
3.32	3.51
3.45	3.33
3.63	3.71
3.51	3.44
3.27	3.29
3.28	3.34
3.43	3.28
3.51	3.63
3.41	3.45
3.50	3.48
3.40	3.51
3.46	3.55
3.53	3.46
3.42	3.48
3.38	3.41
3.43	3.53
3.39	3.44
3.36	3.29
3.41	3.37

The linear association observed in the present study case between LAI_ω and LAI_δ values corresponding to thirty rigid plants colonizing the Italian fully vegetated abandoned ditch analyzed here can be expressed as follows:

$$LAI_\delta = 0.701 * LAI_\omega + 1.052 \tag{5.4}$$

As described in many previous ecohydraulic studies, LAI values retrieved from the field represent support for modeling the eco-hydrodynamic behaviour of constructed and natural vegetated open channels. Thus, given a coefficient of determination R^2 of 0.754, a good correlation was identified by comparing LAI_ω to LAI_δ of thirty rigid plants covering the examined Italian fully vegetated abandoned ditch. The results of the comparative analysis carried out in this subsection study case are

useful for further ecohydraulic and slope stability research, especially for the continuous monitoring of the most relevant hydrodynamic phenomena affecting vegetated waterways colonized by very stiff aquatic vegetation. As pointed out by many notable modeling and experimental ecohydraulic studies and reviews dealing with the prediction of the eco-hydrologic and eco-hydrodynamic parameters characterizing complex aquatic ecosystems, the analysis of the behavior of riverine plants at full scale is a key factor in the environmental engineering management pertaining to vegetated water systems and natural resources in general, especially from the perspective of future severe rainfall and flooding events due to extreme weather conditions related to imminent Climate Change patterns.

5.4.4 Conclusions

According to both forestry engineering models and botanical expectations, the linear relation retrieved in the present study case for obtaining accurate drone-based LAI predictions allows reproducing the actual full-scale morphometric trends of riparian stands to be employed in flume laboratory and field experiments involving various riparian vegetation species covering bed and banks of vegetated waterways. It can be stated that drone-based LAI values associated with hardly rigid riparian vegetation can be effectively considered as good predictors of those obtained from the real-scale scan imagery based on portable gaming-type scan devices. This is a very promising finding, to be taken properly into account for future research dealing with both experimental and modeling Ecohydrology and Ecohydraulics. There is no doubt that considerable advance to the predictive performance of the methodology proposed in this study case can be reached out through the application of high-resolution machine learning approaches, artificial intelligence and soft-computing models in detail, to be applied on ecohydraulic datasets for the estimation of the values of the main traits of riparian vegetation patches and stands over time, affecting both fluid dynamic average patterns and chaotic fluctuations generated by highly three-dimensional vortex shedding and Eco-hydrodynamic wake turbulent structures .

5.5 Scientific work produced

[19] - Giuseppe Francesco Cesare Lama, Mariano Crimaldi, Vittorio Pasquino, Roberta Padulano, and Giovanni B. Chirico. 2021. "Bulk Drag Predictions of Riparian Arundo donax Stands through UAV-Acquired Multispectral Images" Water 13, no. 10: 1333. https://doi.org/10.3390/w13101333

[254] - Mariano Crimaldi and Giuseppe Francesco Cesare Lama. “Impact of Riparian Plants Biomass Assessed by UAV-Acquired Multispectral Images on the Hydrodynamics of Vegetated Streams”. In: European Biomass Conference and Exhibition Proceedings. ETA-Florence Renewable Energies. 2021, pp. 1157–1161. DOI:10.5071/29thEUBCE2021- 4AV.3.6.

[255] - Giuseppe Francesco Cesare Lama and Mariano Crimaldi. “Assessing the Role of Gap Fraction on the Leaf Area Index (LAI) Estimations of Riparian Vegetation Based on Fish-eye Lenses”. In: European Biomass Conference and Exhibition Proceedings. ETA-Florence Renewable Energies. 2021, pp. 1172–1176.
DOI:10.5071/29thEUBCE2021-4AV.3.16.
[256] - Giuseppe Francesco Cesare Lama, Mariano Crimaldi. "Calibration of flow resistance models in vegetated ditches based on UAV remote sensing". In: Proceedings of the 1st International Electronic Conference on Agronomy. 3–17 May 2021. Basel, Switzerland. doi:10.3390/IECAG2021-09728.

Bibliography

[1] Michael Grieves. "Digital twin: manufacturing excellence through virtual factory replication". In: *White paper* 1 (2014), pp. 1–7.

[2] Werner Kritzinger et al. "Digital Twin in manufacturing: A categorical literature review and classification". In: *16th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2018*. Vol. 51. 2018, pp. 1016–1022. DOI: https://doi.org/10.1016/j.ifacol.2018.08.474.

[3] Christos Pylianidis, Sjoukje Osinga, and Ioannis N. Athanasiadis. "Introducing digital twins to agriculture". In: *Computers and Electronics in Agriculture* 184 (2021), p. 105942. ISSN: 0168-1699. DOI: https://doi.org/10.1016/j.compag.2020.105942.

[4] Gaëtan Louarn and Youhong Song. "Two decades of functional–structural plant modelling: now addressing fundamental questions in systems biology and predictive ecology". In: *Annals of Botany* 126.4 (2020), pp. 501–509. ISSN: 0305-7364. DOI: 10.1093/aob/mcaa143.

[5] James W. Jones et al. "Brief history of agricultural systems modeling". In: *Agricultural Systems* 155 (July 2017), pp. 240–254. ISSN: 0308-521X. DOI: 10.1016/j.agsy.2016.05.014.

[6] Andrew J Millar et al. "Practical steps to digital organism models, from laboratory model species to 'Crops in silico'". In: *Journal of Experimental Botany* 70.9 (Apr. 2019), pp. 2403–2418. ISSN: 0022-0957. DOI: 10.1093/jxb/ery435.

[7] Amy Marshall-Colon et al. "Crops In Silico: Generating Virtual Crops Using an Integrative and Multi-scale Modeling Platform". In: *Frontiers in Plant Science* 8 (2017). Publisher: Frontiers. ISSN: 1664-462X. DOI: 10.3389/fpls.2017.00786.

[8] Qi-Long Zhang and Ming-Yong Pang. "A survey of modeling and rendering trees". In: Springer, 2008, pp. 757–764.

[9] Frédéric Boudon, Alexandre Meyer, and Christophe Godin. *Survey on computer representations of trees for realistic and efficient rendering*. Research report 2301. LIRIS UMR CNRS 5205, 2006. URL: https://hal.inria.fr/hal-00830069.

[10] Stanislaw Ulam. "Patterns of growth of figures: Mathematical aspects". In: *Module, proportion, symmetry, rhythm*. New York: G. Braziller, 1966, pp. 64–74.

[11] N. Greene. "Voxel space automata: modeling with stochastic growth processes in voxel space". In: *Proceedings of the 16th annual conference on Computer graphics and interactive techniques*. SIGGRAPH '89. New York, NY, USA: Association for Computing Machinery, July 1989, pp. 175–184. ISBN: 978-0-89791-312-6. DOI: 10.1145/74333.74351.

[12] H. Honda. "Description of the form of trees by the parameters of the tree-like body: Effects of the branching angle and the branch length on the shape of the tree-like body". In: *Journal of theoretical biology* 31.2 (1971), pp. 331–338. ISSN: 0022-5193.

[13] Aristid Lindenmayer. "Mathematical models for cellular interactions in development I. Filaments with one-sided inputs". In: *Journal of theoretical biology* 18.3 (1968), pp. 280–299. ISSN: 0022-5193.

[14] Przemyslaw Prusinkiewicz and Aristid Lindenmayer. *The algorithmic beauty of plants*. Springer Science & Business Media, 2012. ISBN: 1-4613-8476-1.

[15] P.E. Oppenheimer. "Real time design and animation of fractal plants and trees". en. In: *ACM SiGGRAPH Computer Graphics* (1986).

[16] Oliver Deussen and Bernd Lintermann. *Digital design of nature: computer generated plants and organics*. Springer Science & Business Media, 2006. ISBN: 3-540-27104-X.

[17] Binglin Zhu et al. "Quantification of light interception within image-based 3-D reconstruction of sole and intercropped canopies over the entire growth season". In: *Annals of Botany* 126.4 (Sept. 2020), pp. 701–712. ISSN: 0305-7364. DOI: 10.1093/aob/mcaa046.

[18] Yingpu Che et al. "Estimation of maize plant height and leaf area index dynamics using an unmanned aerial vehicle with oblique and nadir photography". In: *Annals of botany* 126.4 (2020), pp. 765–773.

[19] Giuseppe Francesco Cesare Lama et al. "Bulk Drag Predictions of Riparian Arundo donax Stands through UAV-Acquired Multispectral Images". In: *Water* 13.10 (2021), p. 1333.

[20] Sören Pirk et al. "Capturing and animating the morphogenesis of polygonal tree models". In: *ACM Transactions on Graphics (TOG)* 31.6 (2012). Publisher: ACM New York, NY, USA, pp. 1–10.

[21] Miłosz Makowski et al. "Synthetic silviculture: multi-scale modeling of plant ecosystems". In: *ACM Transactions On Graphics* 38.4 (2019), p. 131. ISSN: 0730-0301.

[22] Marcus A. Magnor et al. *Digital representations of the real world: how to capture, model, and render visual reality*. AK Peters/CRC Press, 2015. ISBN: 0-429-16129-8.

[23] Giuseppe Francesco Cesare Lama et al. "Evaluation of Flow Resistance Models Based on Field Experiments in a Partly Vegetated Reclamation Channel". In: *Geosciences* 10.2 (Feb. 2020), p. 47. DOI: 10.3390/geosciences10020047.

[24] M. Henke, W. Kurth, and G.H. Buck-Sorlin. "FSPM-P: towards a general functional-structural plant model for robust and comprehensive model development". In: *Frontiers of Computer Science* 10.6 (2016). DOI: 10.1007/s11704-015-4472-8..

[25] P.-H. Cournède et al. "Structural Factorization of Plants to Compute Their Functional and Architectural Growth". In: *Functional and Architectural Growth* 82.7 (2006). DOI: 10.1177/0037549706069341..

[26] Peter Room, Jim Hanan, and Przemyslaw Prusinkiewicz. "Virtual plants: new perspectives for ecologists, pathologists and agricultural scientists". In: *Trends in Plant Science* 1.1 (1996), pp. 33–38.

[27] Jesús Gumbau et al. "View-dependent pruning for real-time rendering of trees". In: *Computers Graphics* 35.2 (2011), pp. 364–374. ISSN: 0097-8493.

[28] Frédéric Boudon et al. "V-Mango: a functional–structural model of mango tree growth, development and fruit production". In: *Annals of Botany* 126.4 (2020), pp. 745–763. ISSN: 0305-7364. DOI: 10.1093/aob/mcaa089.

[29] J. Vries. *How plants balance competitive growth and defence: an analysis of virtual plants in dynamic interactions*. Wageningen University, 2019.

[30] C. Lecarpentier et al. "WALTer: a three-dimensional wheat model to study competition for light through the prediction of tillering dynamics". In: *Annals of Botany* 123.6 (2019). DOI: 10.1093/aob/mcy226..

[31] Alex Wu et al. "Simulating daily field crop canopy photosynthesis: an integrated software package". In: *Functional Plant Biology* 45.3 (2018), pp. 362–377. DOI: https://doi.org/10.1071/FP17225.

[32] X. Zhou et al. "Presentation of CPlantBox: a whole functional-structural plant model (root and shoot) coupled with a mechanistic resolution of carbon and water flows". In: Nov. 2018, pp. 147–151. DOI: 10.1109/PMA.2018.8611617.

[33] Theodore M. DeJong et al. "Using functional–structural plant models to study, understand and integrate plant development and ecophysiology". In: *Annals of Botany* 108.6 (2011), pp. 987–989. ISSN: 0305-7364. DOI: 10.1093/aob/mcr257.

[34] Christophe Godin and Hervé Sinoquet. "Functional–structural plant modelling". In: *New Phytologist* 166.3 (2005), pp. 705–708. DOI: 10.1111/j.1469-8137.2005.01445.x.

[35] Christophe Godin. "Representing and encoding plant architecture: a review". In: *Annals of forest science* 57.5 (2000), pp. 413–438.

[36] Ayan Chaudhury and Christophe Godin. "Geometry Reconstruction of Plants". Dec. 2019.

[37] Katrin Kahlen and Tsu-Wei Chen. "Predicting Plant Performance Under Simultaneously Changing Environmental Conditions—The Interplay Between Temperature, Light, and Internode Growth". In: *Frontiers in Plant Science* 6 (2015). Publisher: Frontiers. ISSN: 1664-462X. DOI: 10.3389/fpls.2015.01130.

[38] Ningyi Zhang et al. "Disentangling the effects of photosynthetically active radiation and red to far-red ratio on plant photosynthesis under canopy shading: a simulation study using a functional–structural plant model". In: *Annals of botany* 126.4 (2020), pp. 635–646.

[39] Zhuming Lam and Scott A. King. "Simulating tree growth based on internal and environmental factors". In: *Proceedings of the 3rd international conference on Computer graphics and interactive techniques in Australasia and South East Asia*. GRAPHITE '05. Dunedin, New Zealand: Association for Computing Machinery, Nov. 2005, pp. 99–107. ISBN: 978-1-59593-201-3. DOI: 10.1145/1101389.1101406.

[40] Jean-François Barczi et al. "AmapSim: A Structural Whole-plant Simulator Based on Botanical Knowledge and Designed to Host External Functional Models". In: *Annals of Botany* 101.8 (2008), pp. 1125–1138. ISSN: 0305-7364. DOI: 10.1093/aob/mcm194.

[41] Xiao-Ran Zhou et al. "CPlantBox, a whole-plant modelling framework for the simulation of water- and carbon-related processes". In: *in silico Plants* 2.diaa001 (2020). ISSN: 2517-5025. DOI: 10.1093/insilicoplants/diaa001.

[42] Philippe De Reffye et al. "Two decades of research with the GreenLab model in Agronomy". In: *Annals of Botany* (Sept. 2020). DOI: 10.1093/aob/mcaa172.

[43] M. T. Allen, P. Prusinkiewicz, and T. M. DeJong. "Using L-systems for modeling source-sink interactions, architecture and physiology of growing trees: the L-PEACH model". In: *The New Phytologist* 166.3 (June 2005), pp. 869–880. ISSN: 0028-646X.

[44] Brian N Bailey. "Helios: A Scalable 3D Plant and Environmental Biophysical Modeling Framework". In: *Frontiers in plant science* 10 (2019), pp. 1185–1185. ISSN: 1664-462X. DOI: 10.3389/fpls.2019.01185. URL: https://pubmed.ncbi.nlm.nih.gov/31681349.

[45] Ole Kniemeyer, Gerhard Buck-Sorlin, and Winfried Kurth. "GroIMP as a platform for functional-structural modelling of plants". In: *Frontis* (2007), pp. 43–52.

[46] Françoise Lescourret et al. "QualiTree, a virtual fruit tree to study the management of fruit quality. I. Model development". In: *Trees* 25.3 (June 2011), pp. 519–530. ISSN: 1432-2285. DOI: 10.1007/s00468-010-0531-9.

[47] Jean Dauzat et al. “Simulation of ecophysiological processes on 3D virtual stands with the ARCHIMED simulation platform”. In: *2006 second international symposium on plant growth modeling and applications*. 2006, pp. 101–108. DOI: 10.1109/PMA.2006.52.

[48] H. Sinoquet et al. “RATP: a model for simulating the spatial distribution of radiation absorption, transpiration and photosynthesis within canopies: application to an isolated tree crown”. In: *Plant, Cell & Environment* 24.4 (Apr. 2001), pp. 395–406. ISSN: 0140-7791. DOI: 10.1046/j.1365-3040.2001.00694.x.

[49] V. Sarlikioti et al. “How plant architecture affects light absorption and photosynthesis in tomato: towards an ideotype for plant architecture using a functional–structural plant model”. In: *Annals of Botany* 108.6 (2011), pp. 1065–1073. ISSN: 0305-7364. DOI: 10.1093/aob/mcr221.

[50] Christophe Pradal et al. “OpenAlea: a visual programming and component-based software platform for plant modelling”. In: *Functional plant biology* 35.10 (2008), pp. 751–760.

[51] Evelyne Costes et al. “MAppleT: simulation of apple tree development using mixed stochastic and biomechanical models”. In: *Functional Plant Biology* 35.10 (Dec. 2008), pp. 936–950. ISSN: 1445-4416. DOI: 10.1071/FP08081.

[52] Gaetan Louarn, E Lebon, and Jérémie Lecoeur. “"Top-vine", a topiary approach based architectural model to simulate vine canopy structure.” In: *XIV international GESCO viticulture congress*. Geisenheim, Germany, 2005, pp. 464–470.

[53] Jean Bouchon, Philippe De Reffye, and Daniel Barthélémy. “Modélisation et simulation de l’architecture des végétaux”. In: (1997).

[54] M. Jaeger and P. H. De Reffye. “Basic concepts of computer simulation of plant growth”. In: *Journal of Biosciences* 17.3 (Sept. 1992), pp. 275–291. ISSN: 0973-7138. DOI: 10.1007/BF02703154.

[55] Yaffa L Grossman and Theodore M DeJong. “PEACH: a simulation model of reproductive and vegetative growth in peach trees”. In: *Tree Physiology* 14.4 (1994), pp. 329–345.

[56] Radoslaw Karwowski and Przemyslaw Prusinkiewicz. “The L-system-based plant-modeling environment L-studio 4.0”. In: *Proceedings of the 4th international workshop on functional-structural plant models*. 2004, pp. 403–405.

[57] Xavier Le Roux et al. “Carbon-based models of individual tree growth: a critical appraisal”. In: *Annals of Forest Science* 58.5 (2001), pp. 469–506.

[58] TR Sinclair, CE Murphy, and KR Knoerr. “Development and evaluation of simplified models for simulating canopy photosynthesis and transpiration”. In: *Journal of Applied Ecology* (1976), pp. 813–829.

[59] AD Friend. "Modelling canopy CO2 fluxes: are 'big-leaf'simplifications justified?" In: *Global Ecology and Biogeography* 10.6 (2001), pp. 603–619.

[60] MR Raupach and JJ Finnigan. "'Single-layer models of evaporation from plant canopies are incorrect but useful, whereas multilayer models are correct but useless': discuss". In: *Australian Journal of Plant Physiology (Australia)* (1988).

[61] Reinhard Hemmerling et al. "The rule-based language XL and the modelling environment GroIMP illustrated with simulated tree competition". In: *Functional plant biology* 35.10 (2008), pp. 739–750.

[62] Jason Weber and Joseph Penn. "Creation and rendering of realistic trees". In: ACM, 1995, pp. 119–128. ISBN: 0-89791-701-4.

[63] Winfried Kurth, Ole Kniemeyer, and Gerhard Buck-Sorlin. "Relational Growth Grammars – A Graph Rewriting Approach to Dynamical Systems with a Dynamical Structure". In: *Unconventional Programming Paradigms*. Ed. by Jean-Pierre Banâtre et al. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2005, pp. 56–72. ISBN: 978-3-540-31482-0. DOI: 10.1007/11527800_5.

[64] Tomas Plachetka. "POV Ray: persistence of vision parallel raytracer". In: *Proc. of spring conf. on computer graphics, budmerice, slovakia*. Vol. 123. 1998.

[65] Radomír Měch and Przemyslaw Prusinkiewicz. "Visual models of plants interacting with their environment". In: *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. 1996, pp. 397–410.

[66] Radoslaw Karwowski and Przemyslaw Prusinkiewicz. "Design and Implementation of the L+C Modeling Language". In: *4th International Workshop on Rule-Based Programming (in connection with RDP'03, Federated Conference on Rewriting, Deduction and Programming)* 86.2 (2003), pp. 134–152. ISSN: 1571-0661. DOI: 10.1016/S1571-0661(04)80680-7.

[67] Junqi Zhu et al. "A 3-D functional–structural grapevine model that couples the dynamics of water transport with leaf gas exchange". In: *Annals of Botany* 121.5 (Apr. 2018), pp. 833–848. ISSN: 0305-7364. DOI: 10.1093/aob/mcx141.

[68] Francis Hallé, Roelof AA Oldeman, and Philip Barry Tomlinson. *Tropical trees and forests: an architectural analysis*. Springer Science & Business Media, 2012.

[69] Frédéric Boudon et al. "L-Py: an L-system simulation framework for modeling plant architecture development based on a dynamic language". In: *Frontiers in plant science* 3 (2012), p. 76.

[70] Jean Dauzat, Bruno Rapidel, and André Berger. "Simulation of leaf transpiration and sap flow in virtual plants: model description and application to a coffee plantation in Costa Rica". In: *Agricultural and Forest Meteorology* 109.2 (2001), pp. 143 –160. ISSN: 0168-1923. DOI: https://doi.org/10.1016/S0168-1923(01)00236-2.

[71] Pieter Huibert Bram De Visser, Gerie van der Heijden, and Gerhard Buck-Sorlin. "Optimizing illumination in the greenhouse using a 3D model of tomato and a ray tracer". English. In: *Frontiers in Plant Science* 5 (2014). ISSN: 1664-462X. DOI: 10.3389/fpls.2014.00048.

[72] J. Anja Dieleman et al. "Integrating Morphological and Physiological Responses of Tomato Plants to Light Quality to the Crop Level by 3D Modeling". In: *Frontiers in Plant Science* 10 (2019). ISSN: 1664-462X. DOI: 10.3389/fpls.2019.00839.

[73] Gaëtan Louarn, Jérémie Lecoeur, and Eric Lebon. "A Three-dimensional Statistical Reconstruction Model of Grapevine (Vitis vinifera) Simulating Canopy Structure Variability within and between Cultivar/Training System Pairs". In: *Annals of Botany* 101.8 (2008), pp. 1167–1184. ISSN: 0305-7364. DOI: 10.1093/aob/mcm170.

[74] Romain Barillot et al. "L-Pea: an architectural model of pea (Pisum sativum) development". In: *7. International conference on functional structure plant models*. 2013, 371–p.

[75] David Ascher and Mark Lutz. *Learning python*. O'Reilly, 1999.

[76] Bill Councill and George T Heineman. "Definition of a software component and its elements". In: *Component-based software engineering: putting the pieces together* (2001), pp. 5–19.

[77] Céline Leroy et al. "Virtual trees and light capture: a method for optimizing agroforestry stand design". In: *Agroforestry Systems* 77.1 (Sept. 2009), pp. 37–47. ISSN: 1572-9680. DOI: 10.1007/s10457-009-9232-z.

[78] Ning Xia, Aishuang Li, and Weijun Lin. "Simulation and Quantitative Analysis of Branching Patterns of the Plum Tree". In: *Journal of Computer Science Technology Updates* 1.1 (Dec. 2014), pp. 9–18.

[79] Damien Sellier and Thierry Fourcaud. "Crown structure and wood properties: Influence on tree sway and response to high winds". In: *American Journal of Botany* 96.5 (2009), pp. 885–896. ISSN: 1537-2197. DOI: 10.3732/ajb.0800226.

[80] Sébastien Griffon and François De Coligny. "AMAPstudio: an editing and simulation software suite for plants architecture modelling". In: *Ecological Modelling* 290 (2014), pp. 3–10. ISSN: 0304-3800.

[81] Laurent Duchemin et al. "Tree crowns grow into self-similar shapes controlled by gravity and light sensing". In: *Journal of The Royal Society Interface* 15.142 (May 2018), p. 20170976. DOI: 10.1098/rsif.2017.0976.

[82] Miroslava Rakocevic et al. "How density and planting pattern affect coffee plant structure and berry distribution in first production year?" In: *25th International Conference on Coffee Science, ASIC 2014, Armenia, Colombia, 8-13 September 2014*. Association Scientifique Internationale du Café (ASIC), 2014.

[83] Stéphane Momo Takoudjou et al. "Using terrestrial laser scanning data to estimate large tropical trees biomass and calibrate allometric models: A comparison with traditional destructive approach". In: *Methods in Ecology and Evolution* 9.4 (2018), pp. 905–916. ISSN: 2041-210X. DOI: 10.1111/2041-210X.12933.

[84] Federico Magistri et al. "Towards In-Field Phenotyping Exploiting Differentiable Rendering with Self-Consistency Loss". In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. May 2021, pp. 13960–13966. DOI: 10.1109/ICRA48506.2021.9561356.

[85] Mukhtar Ahmed et al. "Nutrient Dynamics and the Role of Modeling". In: *Building Climate Resilience in Agriculture: Theory, Practice and Future Perspective*. Ed. by Wajid Nasim Jatoi et al. Cham: Springer International Publishing, 2022, pp. 297–316. ISBN: 978-3-030-79408-8. DOI: 10.1007/978-3-030-79408-8_19.

[86] Andrea Schnepf et al. "CRootBox: a structural–functional modelling framework for root systems". In: *Annals of botany* 121.5 (2018), pp. 1033–1053.

[87] MengZhen Kang et al. "A virtual plant that responds to the environment like a real one: the case for chrysanthemum". In: *New Phytologist* 195.2 (2012), pp. 384–395. ISSN: 1469-8137. DOI: https://doi.org/10.1111/j.1469-8137.2012.04177.x.

[88] Paul-Henry Cournède et al. "Computing Competition for Light in the GREEN-LAB Model of Plant Growth: A Contribution to the Study of the Effects of Density on Resource Acquisition and Architectural Development". In: *Annals of Botany* 101.8 (Nov. 2007), pp. 1207–1219. ISSN: 0305-7364. DOI: 10.1093/aob/mcm272.

[89] Lu Feng et al. "Comparing an empirical crop model with a functional structural plant model to account for individual variability". In: *European Journal of Agronomy* 53 (2014), pp. 16–27.

[90] Véronique Letort et al. "Quantitative genetics and functional–structural plant growth models: simulation of quantitative trait loci detection for model parameters and application to potential yield optimization". In: *Annals of Botany* 101.8 (2008), pp. 1243–1254.

[91] Charlotte Baey et al. "Parametrization of five classical plant growth models applied to sugar beet and comparison of their predictive capacity on root yield and total biomass". In: *Ecological modelling* 290 (2014). Publisher: Elsevier, pp. 11–20.

[92] David Da Silva et al. "Measuring and modelling seasonal patterns of carbohydrate storage and mobilization in the trunks and root crowns of peach trees". In: *Annals of botany* 114.4 (2014), pp. 643–652.

[93] Gerardo Lopez et al. "Integrating simulation of architectural development and source–sink behaviour of peach trees by incorporating Markov chains and physiological organ function submodels into L-PEACH". In: *Functional Plant Biology* 35.10 (2008), pp. 761–771.

[94] Zhenghao Fei et al. "Enlisting 3D crop models and GANs for more data efficient and generalizable fruit detection". In: *Proceedings of the IEEE/CVF international conference on computer vision (ICCV) workshops*. Oct. 2021, pp. 1269–1277.

[95] María A. Ponce de León and Brian N. Bailey. "A 3D model for simulating spatial and temporal fluctuations in grape berry temperature". In: *Agricultural and Forest Meteorology* 306 (Aug. 2021), p. 108431. ISSN: 0168-1923. DOI: 10.1016/j.agrformet.2021.108431.

[96] William T. Salter, Arjina Shrestha, and Margaret M. Barbour. "Open source 3D phenotyping of chickpea plant architecture across plant development". In: *Plant Methods* 17.1 (Sept. 2021), p. 95. ISSN: 1746-4811. DOI: 10.1186/s13007-021-00795-6.

[97] Christopher Bahr, Dominik Schmidt, and Katrin Kahlen. "Missing Links in Predicting Berry Sunburn in Future Vineyards". In: *Frontiers in Plant Science* 12 (2021), p. 2178. ISSN: 1664-462X. DOI: 10.3389/fpls.2021.715906.

[98] Hongchun Qu et al. "Simulation of carbon-based model for virtual plants as complex adaptive system". In: *Simulation Modelling Practice and Theory* 18.6 (June 2010), pp. 677–695. ISSN: 1569-190X. DOI: 10.1016/j.simpat.2010.01.004.

[99] Jonas R Coussement et al. "Introducing turgor-driven growth dynamics into functional–structural plant models". In: *Annals of Botany* 121.5 (Apr. 2018), pp. 849–861. ISSN: 0305-7364. DOI: 10.1093/aob/mcx144.

[100] Fenni Kang et al. "SUNLAB: A functional–structural model for genotypic and phenotypic characterization of the sunflower crop". In: *Ecological Modelling*. Special Issue of the 4th International Symposium on Plant Growth Modeling, Simulation, Visualization and Applications (PMA'12) 290 (Oct. 2014), pp. 21–33. ISSN: 0304-3800. DOI: 10.1016/j.ecolmodel.2014.02.006.

[101] Michael Henke et al. *Reconstructing leaf growth based on non-destructive digitizing and low-parametric shape evolution for plant modelling over a growth cycle*. Vol. 48. 2014.

[102] Katarína Streit et al. "Impact of geometrical traits on light interception in conifers: Analysis using an FSPM for Scots pine". In: *2016 IEEE International Conference on Functional-Structural Plant Growth Modeling, Simulation, Visualization and Applications (FSPMA)*. Nov. 2016, pp. 194–203. DOI: 10.1109/FSPMA.2016.7818307.

[103] Inigo Auzmendi and Jim Hanan. "Using L-studio to Visualize Data and Modify Plant Architecture for Agronomic Purposes: Visualization and modification of plant architecture with L-studio". In: *2018 6th International Symposium on Plant Growth Modeling, Simulation, Visualization and Applications (PMA)*. Nov. 2018, pp. 43–49. DOI: 10.1109/PMA.2018.8611586.

[104] Nicholas Todd Anderson, Kerry Brian Walsh, and Dvoralai Wulfsohn. "Technologies for Forecasting Tree Fruit Load and Harvest Timing—From Ground, Sky and Time". In: *Agronomy* 11.7 (July 2021), p. 1409. DOI: 10.3390/agronomy11071409.

[105] Junming Zheng et al. "Various distances and orientations influenced the branching traits of Dendrocalamus minor var. amoenus in Dongshan Island, China". In: *Global Ecology and Conservation* 26 (Apr. 2021), e01468. ISSN: 2351-9894. DOI: 10.1016/j.gecco.2021.e01468.

[106] Pierre-Eric Lauri. "Tree architecture and functioning facing multispecies environments: We have gone only halfway in fruit-trees". In: *American Journal of Botany* 108.1 (2021), pp. 3–7. ISSN: 1537-2197. DOI: 10.1002/ajb2.1598.

[107] José M. Mirás-Avalos et al. "QualiTree, a virtual fruit tree to study the management of fruit quality. II. Parameterisation for peach, analysis of growth-related processes and agronomic scenarios". In: *Trees* 25.5 (Oct. 2011), pp. 785–799. ISSN: 1432-2285. DOI: 10.1007/s00468-011-0555-9.

[108] Remi Vezy et al. "Light exchanges in discrete directions as an alternative to raytracing and radiosity". In: Online, 2020. DOI: 10.13140/RG.2.2.13421.20968/1.

[109] Celine Leroy et al. "Simulation of light transmission under Acacia mangium Willd. and Tecton grandis L. canopies and comparison with in situ measurements". In: (2005).

[110] Jorge A Prieto et al. "A functional–structural plant model that simulates whole-canopy gas exchange of grapevine plants (Vitis vinifera L.) under different training systems". In: *Annals of botany* 126.4 (2020), pp. 647–660.

[111] Hong-Ping Yan et al. "A dynamic, architectural plant model simulating resource-dependent growth". In: *Annals of botany* 93.5 (2004), pp. 591–602.

[112] André Lacointe and Peter E. H. Minchin. "A Mechanistic Model to Predict Distribution of Carbon Among Multiple Sinks". In: *Phloem: Methods and Protocols*. Ed. by Johannes Liesche. Methods in Molecular Biology. New York, NY: Springer, 2019, pp. 371–386. ISBN: 978-1-4939-9562-2. DOI: 10.1007/978-1-4939-9562-2_28.

[113] "Investigating the influence of geometrical traits on light interception efficiency of apple trees: A modelling study with MAppleT". In: Oct. 2012, pp. 152–159. DOI: 10.1109/PMA.2012.6524827.

[114] Brian N Bailey. "A reverse ray-tracing method for modelling the net radiative flux in leaf-resolving plant canopy simulations". In: *Ecological Modelling* 368 (2018), pp. 233–245.

[115] Christian A. Gueymard. "Direct solar transmittance and irradiance predictions with broadband models. Part I: detailed theoretical performance assessment". In: *Solar Energy* 74.5 (2003), pp. 355–379. ISSN: 0038-092X. DOI: https://doi.org/10.1016/S0038-092X(03)00195-6.

[116] Thomas N Buckley, Tarryn L Turnbull, and Mark A Adams. "Simple models for stomatal conductance derived from a process model: cross-validation against sap flux data". In: *Plant, cell & environment* 35.9 (2012), pp. 1647–1662.

[117] IR Johnson. "PlantMod: exploring the physiology of plant canopies". In: *IMJ Software, Armidale, NSW, Australia* (2010).

[118] Graham D Farquhar, S von von Caemmerer, and Joseph A Berry. "A biochemical model of photosynthetic CO 2 assimilation in leaves of C 3 species". In: *Planta* 149.1 (1980), pp. 78–90.

[119] Philippe De Reffye et al. "Essai sur les relations entre l'architecture d'un arbre et la grosseur de ses axes végétatifs". In: (1997).

[120] Jonathan Vermeiren et al. "Quantifying the importance of a realistic tomato (Solanum lycopersicum) leaflet shape for 3-D light modelling". In: *Annals of Botany* 126.4 (2020), pp. 661–670. ISSN: 0305-7364. DOI: 10.1093/aob/mcz205.

[121] Jean-Baptiste Durand et al. "Segmentation-based approaches for characterising plant architecture and assessing its plasticity at different scales". In: *FSPM07-5th international workshop on functional-structural plant models*. 2007, pp. 39–1.

[122] Yann Guédon et al. "Identifying and characterizing the ontogenetic component in tree development". In: *5th international workshop on functional-structural plant models*. 2007, pp. 38–1.

[123] Michael Chelle and Bruno Andrieu. "The nested radiosity model for the distribution of light within plant canopies". In: *Ecological Modelling* 111.1 (1998), pp. 75–91.

[124] Jérôme Ngao, Boris Adam, and Marc Saudreau. "Intra-crown spatial variability of leaf temperature and stomatal conductance enhanced by drought in apple tree as assessed by the RATP model". In: *Agricultural and Forest Meteorology* 237-238 (2017), pp. 340–354. ISSN: 0168-1923. DOI: https://doi.org/10.1016/j.agrformet.2017.02.036.

[125] Raphael Perez et al. "Toward a functional-structural model of oil palm accounting for architectural plasticity in response to planting density." In: *FSPM 2020: Towards computable plants. 9th international conference on functional-structural plant models*. 2020.

[126] Z. Liu et al. "Interactive modeling of trees using VR devices". In: *2019 international conference on virtual reality and visualization (ICVRV)*. 2019, pp. 69–75. DOI: 10.1109/ICVRV47840.2019.00020.

[127] Jordan Ubbens et al. "The use of plant models in deep learning: an application to leaf counting in rosette plants". In: *Plant methods* 14.1 (2018), pp. 1–10.

[128] William T. Reeves. "Particle systems—a technique for modeling a class of fuzzy objects". In: *ACM Transactions On Graphics* 2.2 (1983), pp. 91–108. ISSN: 0730-0301.

[129] IDV. *SpeedTree®*. URL: https://store.speedtree.com/2017.

[130] Armando de la Re et al. "Tools for procedural generation of plants in virtual scenes". In: Springer, 2009, pp. 801–810.

[131] W. Diestel. *Arbaro - tree generation for povray*. URL: http://arbaro.sourceforge.net/2015.

[132] OnyxComputing. *OnyxTree*. 2017. URL: http://www.onyxtree.com/..

[133] XFrog. *XFrog Plants*. URL: https://xfrog.com/category/xfrogplants.html..

[134] Laubwerk. *Plants Kit 14®*. 2019. URL: https://www.laubwerk.com/products/.

[135] Sören Pirk et al. "Plastic trees: interactive self-adapting botanical tree models". In: *ACM Transactions on Graphics (TOG)* 31.4 (2012), pp. 1–10.

[136] F12 Wybren van Keulen. *TheGrove3D©*. 2019.

[137] Simulistics. *Simile*. Apr. 2021. URL: https://www.simulistics.com/ (visited on 04/10/2021).

[138] Mariano Crimaldi, Edoardo Pasolli, and Francesco Giannino. "Use of process-based system dynamic models to generate biologically aware 3D trees". In: *9th International Conference on Functional-Structural Plant Models: FSPM2020*. Hannover: Institute of Horticultural Production Systems, Oct. 2020, pp. 117–118.

[139] Mariano Crimaldi, Fabrizio Cartenì, and Francesco Giannino. "VISmaF: Synthetic tree for immersive virtual visualization in smart farming". In: *1st International Electronic Conference on Agronomy session Precision and Digital Agriculture*. Basel, Switzerland: MDPI, May 2021. DOI: doi:10.3390/IECAG2021-09880.

[140] Mariano Crimaldi, Fabrizio Cartenì, and Francesco Giannino. "VISmaF: Synthetic Tree for Immersive Virtual Visualization in Smart Farming. Part I: Scientific Background Review and Model Proposal". In: *Agronomy* 11.12 (Dec. 2021), p. 2458. DOI: 10.3390/agronomy11122458.

[141] Nicholas Todd Anderson, Kerry Brian Walsh, and Dvoralai Wulfsohn. "Technologies for Forecasting Tree Fruit Load and Harvest Timing—From Ground, Sky and Time". In: *Agronomy* 11.7 (July 2021), p. 1409. DOI: 10.3390/agronomy11071409.

[142] Federico Magistri et al. "Towards In-Field Phenotyping Exploiting Differentiable Rendering with Self-Consistency Loss". In: May 2021, pp. 13960–13966. DOI: 10.1109/ICRA48506.2021.9561356.

[143] Sören Pirk et al. "Plastic trees: interactive self-adapting botanical tree models". In: *ACM Transactions on Graphics (TOG)* 31.4 (2012), pp. 1–10.

[144] Christian Ernest Vincenot et al. "Theoretical considerations on the combined use of System Dynamics and individual-based modeling in ecology". In: *Ecological Modelling* 222.1 (2011), pp. 210–218.

[145] Unity Technologies. *Unity© Game Engine*. 2021. URL: www.unity3d.com.

[146] C. Runge. "Ueber die numerische Auflösung von Differentialgleichungen". In: *Mathematische Annalen* 46.2 (June 1895), pp. 167–178. ISSN: 1432-1807. DOI: 10.1007/BF01446807.

[147] Tomas Akenine-Moller, Eric Haines, and Naty Hoffman. *Real-time rendering*. AK Peters/crc Press, 2019.

[148] ITU-R. *BT.709 : Parameter values for the HDTV standards for production and international programme exchange*. 2015. URL: https://www.itu.int/rec/R-REC-BT.709-6-201506-I/en.

[149] Mariano Crimaldi, Fabrizio Cartenì, and Francesco Giannino. "VISmaF: Synthetic Tree for Immersive Virtual Visualization in Smart Farming. Part II: model development, results and potential agronomic applications". In: *SUBMITTED to Agronomy* (2022).

[150] E-C Oerke. "Crop losses to pests". In: *The Journal of Agricultural Science* 144.1 (2006), pp. 31–43.

[151] A. C. Guglielmini, A. M. C. Verdú, and E. H. Satorre. "Competitive ability of five common weed species in competition with soybean". In: *International Journal of Pest Management* 63.1 (Jan. 2017), pp. 30–36. ISSN: 0967-0874. DOI: 10.1080/09670874.2016.1213459.

[152] Harmanpreet Kaur, Gursharnpreet Singh Brar, and Poonam Pandurang Shete. "A Review on Different Weed Management Approaches". In: *International Journal of Current Microbiology and Applied Sciences* 8.08 (Aug. 2019), pp. 2854–2859. ISSN: 23197692, 23197706. DOI: 10.20546/ijcmas.2019.808.328.

[153] Santosh Korav et al. "Weed Pressure on Growth and Yield of Groundnut (Arachis hypogaea L.) in Meghalaya, India". In: *International Journal of Current Microbiology and Applied Sciences* 7.03 (Mar. 2018), pp. 2852–2858. ISSN: 23197692, 23197706. DOI: 10.20546/ijcmas.2018.703.328.

[154] Curtis R. Thompson, J. Anita Dille, and Dallas E. Peterson. "Weed Competition and Management in Sorghum". In: *Sorghum*. John Wiley & Sons, Ltd, 2019, pp. 347–360. ISBN: 978-0-89118-628-1. DOI: 10.2134/agronmonogr58.c15.

[155] Ali Zohaib, Tasawer Abbas, and Tahira Tabassum. "Weeds Cause Losses in Field Crops through Allelopathy". In: *Notulae Scientia Biologicae* 8.1 (Mar. 2016), pp. 47–56. ISSN: 2067-3264. DOI: 10.15835/nsb819752.

[156] V. Cirillo et al. "Crop-weed interactions in saline environments". In: *European Journal of Agronomy* 99 (Sept. 2018), pp. 51–61. ISSN: 1161-0301. DOI: 10.1016/j.eja.2018.06.009.

[157] Amber L. Hauvermale and Marwa N. M. E. Sanad. "Phenological Plasticity of Wild and Cultivated Plants". In: *Plant Communities and Their Environment* (May 2019). DOI: 10.5772/intechopen.85070.

[158] Longbiao Guo et al. "Genomic Clues for Crop–Weed Interactions and Evolution". In: *Trends in Plant Science* 23.12 (Dec. 2018), pp. 1102–1115. ISSN: 1360-1385. DOI: 10.1016/j.tplants.2018.09.009.

[159] Poonam Roshan, Aditya Kulshreshtha, and Vipin Hallan. "Global weed-infecting geminiviruses". In: *Geminiviruses*. Springer, 2019, pp. 103–121.

[160] EUSROSTAT. *Statistics | Eurostat*. URL: https://ec.europa.eu/eurostat/databrowser/view/aei_fm_salpest09/default/table?lang=en (visited on 11/27/2020).

[161] D Kent Shannon, David E Clay, and Kenneth A Sudduth. "An introduction to precision agriculture". In: *Precision agriculture basics* (2018), pp. 1–12.

[162] Anne-Katrin Mahlein. "Plant disease detection by imaging sensors–parallels and specific demands for precision agriculture and plant phenotyping". In: *Plant disease* 100.2 (2016), pp. 241–251.

[163] Said Nawar et al. "Delineation of soil management zones for variable-rate fertilization: A review". In: *Advances in agronomy* 143 (2017), pp. 175–245.

[164] Prem Veer Gautam et al. "Mechatronics application in precision sowing: A review". In: *Int. J. Curr. Microbiol. App. Sci* 8.4 (2019), pp. 1793–1807.

[165] Anna Chlingaryan, Salah Sukkarieh, and Brett Whelan. "Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: A review". In: *Computers and electronics in agriculture* 151 (2018), pp. 61–69.

[166] UM Rao Mogili and B B V L Deepak. "Review on Application of Drone Systems in Precision Agriculture". In: *Procedia Computer Science*. International Conference on Robotics and Smart Manufacturing (RoSMa2018) 133 (Jan. 2018), pp. 502–509. ISSN: 1877-0509. DOI: 10.1016/j.procs.2018.07.063.

[167] Kowligi R Krishna. *Agricultural drones: a peaceful pursuit*. CRC Press, 2018.

[168] Joseph E. Hunter et al. "Integration of remote-weed mapping and an autonomous spraying unmanned aerial vehicle for site-specific weed management". In: *PEST MANAGEMENT SCIENCE* 76.4 (Apr. 2020), pp. 1386–1392. ISSN: 1526-498X. DOI: 10.1002/ps.5651.

[169] Kavhiza Nyasha John et al. "Weed mapping technologies in discerning and managing weed infestation levels of farming systems". In: *Research on Crops* 21.1 (2020), pp. 93–98.

[170] Philipp Lottes et al. "UAV-based crop and weed classification for smart farming". In: *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2017, pp. 3024–3031.

[171] Francisca López-Granados et al. "Early season weed mapping in sunflower using UAV technology: variability of herbicide treatment maps against weed thresholds". In: *Precision Agriculture* 17.2 (2016), pp. 183–199.

[172] Francisco Manuel Jiménez-Brenes et al. "Automatic UAV-based detection of Cynodon dactylon for site-specific vineyard management". In: *PloS one* 14.6 (2019), e0218132.

[173] Nikos Alexandratos and Jelle Bruinsma. "World agriculture towards 2030/2050: the 2012 revision". In: ESA Working Papers 12-03 (2012). DOI: 10.22004/ag.econ.288998.

[174] Pasquale Daponte et al. "A review on the use of drones for precision agriculture". In: *IOP Conference Series: Earth and Environmental Science* 275 (May 2019), p. 012022. ISSN: 1755-1315. DOI: 10.1088/1755-1315/275/1/012022.

[175] Panagiotis Radoglou-Grammatikis et al. "A compilation of UAV applications for precision agriculture". In: *Computer Networks* 172 (May 2020), p. 107148. ISSN: 1389-1286. DOI: 10.1016/j.comnet.2020.107148.

[176] Adriano Mancini, Emanuele Frontoni, and Primo Zingaretti. "Satellite and UAV data for Precision Agriculture Applications". In: *2019 INTERNATIONAL CONFERENCE ON UNMANNED AIRCRAFT SYSTEMS (ICUAS' 19)*. International Conference on Unmanned Aircraft Systems. 2019, pp. 491–497. ISBN: 978-1-72810-332-7.

[177] A. Brook et al. "A smart multiple spatial and temporal resolution system to support precision agriculture from satellite images: Proof of concept on Aglianico vineyard". In: *Remote Sensing of Environment* 240 (Apr. 2020), p. 111679. ISSN: 0034-4257. DOI: 10.1016/j.rse.2020.111679.

[178] Adam C. Watts, Vincent G. Ambrosia, and Everett A. Hinkley. "Unmanned Aircraft Systems in Remote Sensing and Scientific Research: Classification and Considerations of Use". In: *Remote Sensing* 4.6 (June 2012), pp. 1671–1692. DOI: 10.3390/rs4061671.

[179] Salvatore Manfreda et al. "On the Use of Unmanned Aerial Systems for Environmental Monitoring". In: *Remote Sensing* 10.4 (Apr. 2018). DOI: 10.3390/rs10040641.

[180] Karen Anderson and Kevin J. Gaston. "Lightweight unmanned aerial vehicles will revolutionize spatial ecology". In: *Frontiers in Ecology and the Environment* 11.3 (2013), pp. 138–146. ISSN: 1540-9309. DOI: 10.1890/120150.

[181] Ken Whitehead and Chris H. Hugenholtz. "Remote sensing of the environment with small unmanned aircraft systems (UASs), part 1: a review of progress and challenges". In: *Journal of Unmanned Vehicle Systems* 02.03 (July 2014), pp. 69–85. ISSN: 2291-3467. DOI: 10.1139/juvs-2014-0006.

[182] J. Torres-Sánchez et al. "Multi-temporal mapping of the vegetation fraction in early-season wheat fields using images from UAV". In: *Computers and Electronics in Agriculture* 103 (Apr. 2014), pp. 104–113. ISSN: 0168-1699. DOI: 10.1016/j.compag.2014.02.009.

[183] Ramia Jannoura et al. "Monitoring of crop biomass using true colour aerial photographs taken from a remote controlled hexacopter". In: *Biosystems Engineering* 129 (Jan. 2015), pp. 341–351. ISSN: 1537-5110. DOI: 10.1016/j.biosystemseng.2014.11.007.

[184] Takeshi Motohka et al. "Applicability of Green-Red Vegetation Index for Remote Sensing of Vegetation Phenology". In: *Remote Sensing* 2.10 (Oct. 2010), pp. 2369–2387. DOI: 10.3390/rs2102369.

[185] Jinru Xue and Baofeng Su. *Significant Remote Sensing Vegetation Indices: A Review of Developments and Applications*. Review Article. May 2017. DOI: https://doi.org/10.1155/2017/1353691.

[186] Mariano Crimaldi et al. "Neural Network Algorithms for Real Time Plant Diseases Detection Using UAVs". In: *Innovative Biosystems Engineering for Sustainable Agriculture, Forestry and Food Production*. Ed. by Antonio Coppola et al. Lecture Notes in Civil Engineering. Cham: Springer International Publishing, 2020, pp. 827–835. ISBN: 978-3-030-39299-4. DOI: 10.1007/978-3-030-39299-4_89.

[187] Michael Pflanz, Michael Schirrmann, and Henning Nordmeyer. "Drone based weed monitoring with an image feature classifier". In: *28TH GERMAN CONFERENCE ON WEED BIOLOGY AND WEED CONTROL*. Ed. by Nordmeyer, H and Ulber, L. Vol. 458. Julius-Kuhn-Archiv. JULIUS KUHN-INST, 2018, pp. 379–384. ISBN: 978-3-95547-054-8. DOI: 10.5073/jka.2018.458.056.

[188] Aaron Etienne and Dharmendra Saraswat. "Machine Learning Approaches to Automate Weed Detection by UAV based sensors". In: *AUTONOMOUS AIR AND GROUND SENSING SYSTEMS FOR AGRICULTURAL OPTIMIZATION AND PHENOTYPING IV*. Ed. by Thomasson, JA and McKee, M and

Moorhead, RJ. Vol. 11008. Proceedings of SPIE. 2019. ISBN: 978-1-5106-2682-9. DOI: 10.1117/12.2520536.

[189] Shubo Wang et al. "A Deep-Learning- Based Low-Altitude Remote Sensing Algorithm for Weed Classification in Ecological Irrigation Area". In: *INTELLIGENT TECHNOLOGIES AND APPLICATIONS, INTAP 2018*. Ed. by Bajwa, IS and Kamareddine, F and Costa, A. Vol. 932. Communications in Computer and Information Science. 2019, pp. 451–460. ISBN: 978-981-13-6052-7. DOI: 10.1007/978-981-13-6052-7_39.

[190] M. Dian Bah, Adel Hafiane, and Raphael Canals. "Deep Learning with Unsupervised Data Labeling for Weed Detection in Line Crops in UAV Images". In: *emote Sensing* 10.11 (Nov. 2018). DOI: 10.3390/rs10111690.

[191] David P. Hughes and Marcel Salathe. "An open access repository of images on plant health to enable the development of mobile disease diagnostics". In: *arXiv:1511.08060 [cs]* (Apr. 2016). URL: http://arxiv.org/abs/1511.08060 (visited on 07/08/2020).

[192] Davinder Singh et al. "PlantDoc: A Dataset for Visual Plant Disease Detection". In: *Proceedings of the 7th ACM IKDD CoDS and 25th COMAD*. CoDS COMAD 2020. Hyderabad, India: Association for Computing Machinery, Jan. 2020, pp. 249–253. ISBN: 978-1-4503-7738-6. DOI: 10.1145/3371158.3371196.

[193] Inkyu Sa et al. "WeedMap: A Large-Scale Semantic Weed Mapping Framework Using Aerial Multispectral Imaging and Deep Neural Network for Precision Farming". In: *Remote Sensing* 10.9 (2018). DOI: 10.3390/rs10091423.

[194] Patrick Wspanialy, Justin Brooks, and Medhat Moussa. "An Image Labeling Tool and Agricultural Dataset for Deep Learning". In: *arXiv:2004.03351 [cs, eess]* (Apr. 2020).

[195] Andrea S. Laliberte et al. "Multispectral Remote Sensing from Unmanned Aircraft: Image Processing Workflows and Applications for Rangeland Environments". In: *Remote Sensing* 3.11 (Nov. 2011), pp. 2529–2551. DOI: 10.3390/rs3112529.

[196] Oscar Barrero and Sammy A. Perdomo. "RGB and multispectral UAV image fusion for Gramineae weed detection in rice fields". In: *Precision Agriculture* 19.5 (2018), pp. 809–822. ISSN: 1385-2256. DOI: 10.1007/s11119-017-9558-x.

[197] Sebastian Candiago et al. "Evaluating Multispectral Images and Vegetation Indices for Precision Farming Applications from UAV Images". In: *Remote Sensing* 7.4 (Apr. 2015), pp. 4026–4047. ISSN: 2072-4292. DOI: 10.3390/rs70404026.

[198] Anna Brook and Eyal Ben-Dor. "Supervised Vicarious Calibration (SVC) of Multi-Source Hyperspectral Remote-Sensing Data". In: *Remote Sensing* 7.5 (May 2015), pp. 6196–6223. DOI: 10.3390/rs70506196.

[199] Telmo Adão et al. "Hyperspectral Imaging: A Review on UAV-Based Sensors, Data Processing and Applications for Agriculture and Forestry". In: *Remote Sensing* 9.11 (Nov. 2017), p. 1110. DOI: 10.3390/rs9111110.

[200] Samuel C. Hassler and Fulya Baysal-Gurel. "Unmanned Aircraft System (UAS) Technology and Applications in Agriculture". English. In: *Agronomy* 9.10 (2019). DOI: 10.3390/agronomy9100618.

[201] Dimosthenis C Tsouros, Stamatia Bibi, and Panagiotis G Sarigiannidis. "A review on UAV-based applications for precision agriculture". In: *Information* 10.11 (2019), p. 349.

[202] Wouter H. Maes and Kathy Steppe. "Perspectives for Remote Sensing with Unmanned Aerial Vehicles in Precision Agriculture". In: *TRENDS IN PLANT SCIENCE* 24.2 (Feb. 2019), pp. 152–164. ISSN: 1360-1385. DOI: 10.1016/j.tplants.2018.11.007.

[203] Camilo Franco et al. "Automatic Detection of Thistle-Weeds in Cereal Crops from Aerial RGB Images". In: ed. by Medina, J and OjedaAciego, M and Verdegay, JL and Perfilieva, I and BouchonMeunier, B and Yager, RR. Vol. 855. Communications in Computer and Information Science. 2018, pp. 441–452. ISBN: 978-3-319-91479-4 978-3-319-91478-7. DOI: 10.1007/978-3-319-91479-4_37.

[204] Yanbo Huang et al. "UAV Low-Altitude Remote Sensing for Precision Weed Management". In: *Weed Technology* 32.1 (Feb. 2018), pp. 2–6. ISSN: 0890-037X. DOI: 10.1017/wet.2017.89.

[205] Marine Louargant et al. "Weed detection by UAV: Simulation of the impact of spectral mixing in multispectral images". In: *Precision Agriculture* 18.6 (2017), pp. 932–951.

[206] Francisca López-Granados et al. "Using remote sensing for identification of late-season grass weed patches in wheat". In: *Weed Science* 54.2 (2006), pp. 346–353.

[207] JT Sanders et al. "Weed species differentiation using spectral reflectance land image classification". In: *Advanced Environmental, Chemical, and Biological Sensing Technologies XV*. Vol. 11007. International Society for Optics and Photonics. 2019, 110070P.

[208] Ana I De Castro et al. "An automatic random forest-OBIA algorithm for early weed mapping between and within crop rows using UAV imagery". In: *Remote Sensing* 10.2 (2018), p. 285.

[209] David J Gibson, Bryan G Young, and Andrew J Wood. "Can weeds enhance profitability? Integrating ecological concepts to address crop-weed competition and yield quality". In: *Journal of Ecology* 105.4 (2017), pp. 900–904.

[210] KDL Millar et al. "Evaluation of physiological parameters for the prediction of seed yield and quality for soybean (Glycine max) plants grown in the presence of weed competition". In: *Plant Biosystems* 145.1 (2011), pp. 1–11.

[211] Guillaume Adeux et al. "Mitigating crop yield losses through weed diversity". In: *Nature Sustainability* 2.11 (2019), pp. 1018–1026.

[212] Tanha Talaviya et al. "Implementation of artificial intelligence in agriculture for optimisation of irrigation and application of pesticides and herbicides". In: *Artificial Intelligence in Agriculture* 4 (2020), pp. 58–73.

[213] Bruno Lanz, Simon Dietz, and Tim Swanson. "The expansion of modern agriculture and global biodiversity decline: an integrated assessment". In: *Ecological Economics* 144 (2018), pp. 260–277.

[214] Jan Willem Erisman et al. "Agriculture and biodiversity: a better balance benefits both". In: *AIMS Agriculture and Food* 1.2 (2016), pp. 157–174.

[215] Marco Esposito et al. "Drone and sensor technology for sustainable weed management: a review". In: *Chemical and Biological Technologies in Agriculture* 8.1 (2021), pp. 1–11.

[216] Marco Esposito et al. "Sustainable Weed Management: Modern Approaches and New Perspectives". In: *Sustainable management of cropping systems*. Bari, 2020, p. 24.

[217] Francesco De Paola and Antonio Ranucci. "Analysis of spatial variability for stormwater capture tank assessment". In: *Irrigation and Drainage* 61.5 (2012), pp. 682–690.

[218] Roberta Padulano, Alfredo Reder, and Guido Rianna. "An ensemble approach for the analysis of extreme rainfall under climate change in Naples (Italy)". In: *Hydrological Processes* 33.14 (2019), pp. 2020–2036.

[219] V García-Gamero et al. "Factors controlling the asymmetry of soil moisture and vegetation dynamics in a hilly Mediterranean catchment". In: *Journal of Hydrology* 598 (2021), p. 126207.

[220] Kaisa Västilä and Juha Järvelä. "Modeling the flow resistance of woody vegetation using physically based properties of the foliage and stem". In: *Water Resources Research* 50.1 (2014), pp. 229–245. ISSN: 1944-7973. DOI: https://doi.org/10.1002/2013WR013819.

[221] Carlo Gualtieri et al. "Bedform Morphology in the Area of the Confluence of the Negro and Solimões-Amazon Rivers, Brazil". In: *Water* 12.6 (2020), p. 1630.

[222] Jonas Schoelynck and Eric Struyf. "Silicon in aquatic vegetation". In: *Functional Ecology* 30.8 (2016), pp. 1323–1330.

[223] Giuseppe Francesco Cesare Lama and Giovanni Battista Chirico. "Effects of reed beds management on the hydrodynamic behaviour of vegetated open channels". In: *2020 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor)*. IEEE. 2020, pp. 149–154.

[224] Nathalie JJ Breda. "Ground-based measurements of leaf area index: a review of methods, instruments and current controversies". In: *Journal of experimental botany* 54.392 (2003), pp. 2403–2417.

[225] Dong Sop Rhee et al. "Hydraulic resistance of some selected vegetation in open channel flows". In: *River research and applications* 24.5 (2008), pp. 673–687.

[226] A Cáceres-Euse et al. "Simplified Model for Oscillatory Flow-Submerged Canopy Interaction". In: *Journal of Hydraulic Engineering* 146.11 (2020), p. 04020073.

[227] Ling Zhu and Qin Chen. "Numerical modeling of surface waves over submerged flexible vegetation". In: *Journal of Engineering Mechanics* 141.8 (2015), A4015001.

[228] Navid Tahvildari. "Numerical modeling of the interactions between nonlinear waves and arbitrarily flexible vegetation". In: *Coastal Engineering Proceedings* 35 (2017), pp. 32–32.

[229] Yuri Taddia et al. "Multispectral UAV monitoring of submerged seaweed in shallow water". In: *Applied Geomatics* 12.1 (2020), pp. 19–34.

[230] Zhan Li et al. "Discriminating treed and non-treed wetlands in boreal ecosystems using time series Sentinel-1 data". In: *International Journal of Applied Earth Observation and Geoinformation* 85 (2020), p. 102007.

[231] Laura Chasmer et al. "Remote sensing of boreal wetlands 2: methods for evaluating boreal wetland ecosystem state and drivers of change". In: *Remote Sensing* 12.8 (2020), p. 1321.

[232] Javier Bustamante et al. "Hyperspectral sensors as a management tool to prevent the invasion of the exotic cordgrass spartina densiflora in the doñana wetlands". In: *Remote Sensing* 8.12 (2016), p. 1001.

[233] Vahid Etminan, Ryan J Lowe, and Marco Ghisalberti. "A new model for predicting the drag exerted by vegetation canopies". In: *Water Resources Research* 53.4 (2017), pp. 3179–3196.

[234] Aleem Khaliq et al. "Comparison of Satellite and UAV-Based Multispectral Imagery for Vineyard Variability Assessment". In: *Remote Sensing* 11.4 (Jan. 2019), p. 436. DOI: 10.3390/rs11040436.

[235] G. F. C. Lama et al. "Velocity uncertainty quantification based on Riparian vegetation indices in open channels colonized by Phragmites australis". In: *Journal of Ecohydraulics* 0.0 (June 2021), pp. 1–6. ISSN: 2470-5357. DOI: 10.1080/24705357.2021.1938255.

[236] Baabak Mamaghani and Carl Salvaggio. "Multispectral Sensor Calibration and Characterization for sUAS Remote Sensing". In: *Sensors* 19.20 (Jan. 2019), p. 4453. DOI: 10.3390/s19204453.

[237] Teemu Hakala et al. "Direct Reflectance Measurements from Drones: Sensor Absolute Radiometric Calibration and System Tests for Forest Reflectance Characterization". In: *Sensors* 18.5 (May 2018), p. 1417. DOI: 10.3390/s18051417.

[238] Desta Ekaso, Francesco Nex, and Norman Kerle. "Accuracy assessment of real-time kinematics (RTK) measurements on unmanned aerial vehicles (UAV) for direct geo-referencing". In: *Geo-spatial information science* 23.2 (2020), pp. 165–181.

[239] Naoyuki Hashimoto et al. "Simulation of Reflectance and Vegetation Indices for Unmanned Aerial Vehicle (UAV) Monitoring of Paddy Fields". In: *Remote Sensing* 11.18 (Jan. 2019), p. 2119. DOI: 10.3390/rs11182119.

[240] Attilio Di Nisio et al. "Fast Detection of Olive Trees Affected by Xylella Fastidiosa from UAVs Using Multispectral Imaging". In: *Sensors* 20.17 (Jan. 2020), p. 4915. DOI: 10.3390/s20174915.

[241] Songyang Li et al. "Combining Color Indices and Textures of UAV-Based Digital Imagery for Rice LAI Estimation". In: *Remote Sensing* 11.15 (Jan. 2019), p. 1763. DOI: 10.3390/rs11151763.

[242] J Martínez-Guanter et al. "Estimation of the leaf area index in maize based on UAV imagery using deep learning techniques". In: *Precision agriculture'19*. Wageningen Academic Publishers, 2019, p. 1304.

[243] Richard Arsenault, François Brissette, and Jean-Luc Martel. "The hazards of split-sample validation in hydrological model calibration". In: *Journal of Hydrology* 566 (Nov. 2018). ISSN: 0022-1694. DOI: 10.1016/j.jhydrol.2018.09.027.

[244] Jose A Jimenez-Berni et al. "High throughput determination of plant height, ground cover, and above-ground biomass in wheat with LiDAR". In: *Frontiers in plant science* 9 (2018), p. 237.

[245] Francesco Fabbrini et al. "Characterization of phenology, physiology, morphology and biomass traits across a broad Euro-Mediterranean ecotypic panel of the lignocellulosic feedstock Arundo donax". In: *GCB Bioenergy* 11.1 (2019), pp. 152–170. ISSN: 1757-1707. DOI: 10.1111/gcbb.12555.

[246] M. Fagnano et al. "Agronomic and Environmental Impacts of Giant Reed (Arundo donax L.): Results from a Long-Term Field Experiment in Hilly Areas Subject to Soil Erosion". In: *BioEnergy Research* 8.1 (Mar. 2015), pp. 415–422. ISSN: 1939-1242. DOI: 10.1007/s12155-014-9532-7.

[247] Chang-Wei Tan et al. "Quantitative monitoring of leaf area index in wheat of different plant types by integrating NDVI and Beer-Lambert law". In: *Scientific Reports* 10.1 (Jan. 2020), p. 929. ISSN: 2045-2322. DOI: 10.1038/s41598-020-57750-z.

[248] Mitul Luhar and Heidi M. Nepf. "From the blade scale to the reach scale: A characterization of aquatic vegetative drag". In: *35th Year Anniversary Issue* 51 (Jan. 2013), pp. 305–316. ISSN: 0309-1708. DOI: 10.1016/j.advwatres.2012.02.002.

[249] Xiaoxia Zhang and Heidi Nepf. "Flow-induced reconfiguration of aquatic plants, including the impact of leaf sheltering". In: *Limnology and Oceanography* 65.11 (2020), pp. 2697–2712. DOI: https://doi.org/10.1002/lno.11542.

[250] Davide Vettori and TI Marjoribanks. "Temporal variability and within-plant heterogeneity in blade biomechanics regulate flow-seagrass interactions of Zostera marina". In: *Water Resources Research* 57.3 (2021).

[251] Luke A Brown, Booker O Ogutu, and Jadunandan Dash. "Estimating forest leaf area index and canopy chlorophyll content with Sentinel-2: An evaluation of two hybrid retrieval algorithms". In: *Remote Sensing* 11.15 (2019), p. 1752.

[252] G Sofia et al. "Flood dynamics in urbanised landscapes: 100 years of climate and humans' interaction". In: *Scientific reports* 7.1 (2017), pp. 1–12.

[253] Giuseppe Francesco Cesare Lama and Mariano Crimaldi. "Riparian Plants' Morphometry Derived by RGB+ Structured-light 3D Scanning within Real Vegetated Flows". In: (2021).

[254] M. Crimaldi and G. F. C. Lama. "Impact of Riparian Plants Biomass Assessed by UAV-Acquired Multispectral Images on the Hydrodynamics of Vegetated Streams". In: *European Biomass Conference and Exhibition Proceedings*. 2021, pp. 1157–1161. DOI: 10.5071/29thEUBCE2021-4AV.3.6.

[255] G. F. C. Lama and M. Crimaldi. "Assessing the Role of Gap Fraction on the Leaf Area Index (LAI) Estimations of Riparian Vegetation Based on Fisheye Lenses". In: *European Biomass Conference and Exhibition Proceedings*. ETA-Florence Renewable Energies. 2021, pp. 1172–1176. DOI: 10.5071/29thEUBCE2021-4AV.3.16.

[256] Giuseppe Francesco Cesare Lama and Mariano Crimaldi. "Calibration of Flow Resistance Models in Vegetated Ditches Based on UAV Remote Sensing". In: *Presented at the 1st International Electronic Conference on Agronomy*. Vol. 3. 2021, p. 17.