

Ciudad Politécnica de la Innovación



# WIICT2023

Proceedings of the  
Workshop on Innovation on Information  
and Communication Technologies 2023

 Institute  
ITACA  
Information and Communication Technologies

Editors:  
Dr. Carlos Fernandez-Llatas  
Dra. Maria Guillen



## Committees

### Organizing Committee

- Chair: Maria Guillem, Universitat Politècnica de València, Spain
- Alberto Bonastre, Universitat Politècnica de València, Spain
- Jose Manuel Catala, Universitat Politècnica de València, Spain
- Carlos Fernández-Llatas, Universitat Politècnica de València, Spain
- Jose Mariano Dahoui, Universitat Politècnica de València, Spain

### Scientific Committee

- Chair: [Carlos Fernandez-Llatas](#), Universitat Politècnica de València, Spain
- Paulo de Carvalho, University of Coimbra, Portugal
- Anna-Maria Bianchi, Politecnico di Milano, Italy
- Jorge Munoz-Gama, Pontificia Universidad Catolica de Chile
- Fernando Seoane, Karolinska Institutet
- Jorge Henriques, University of Coimbra, Portugal
- Cenk Demiroglu, Ozyegin University, Turkey
- Johan Gustav Bellika, National Center of Telemedicine, Norway
- Yigzaw Kassaye Yitbarek, University of Tromso, Norway
- Raymundo Barrales, Universidad Autónoma Metropolitana de México
- Onur Dogan, Istanbul Technical University,, Turkey
- [Frank Y. Li](#), University of Agder, Norway
- Pilar Sala, MySphera., Spain
- Alvaro Martinez, MySphera, Spain
- Jose Carlos Campelo, Universitat Politècnica de València, Spain
- Juan Vicente Capella, Universitat Politècnica de València, Spain
- Antonio Mocholí, Universitat Politècnica de València, Spain
- Sara Blanc, Universitat Politècnica de València, Spain
- Juan-Carlos Baraza-Calvo, Universitat Politècnica de València, Spain
- Juan Carlos Ruiz, Universitat Politècnica de València, Spain
- Joaquin Gracia, Universitat Politècnica de València, Spain
- David De Andres, Universitat Politècnica de València, Spain
- Ricardo Mercado, Universitat Politècnica de València, Spain
- Lenin Lemus, Universitat Politècnica de València, Spain
- Vicente Traver, Universitat Politècnica de València, Spain
- Antonio Martinez-Millana, Universitat Politècnica de València, Spain
- Jose Luis Bayo-Monton, Universitat Politècnica de València, Spain
- Juan Miguel García-Gomez, Universitat Politècnica de València, Spain
- Elies Fuster, Universitat Politècnica de València, Spain
- Carlos Saez, Universitat Politècnica de València, Spain
- Ángel Perles, Universitat Politècnica de València, Spain
- Luis José Saiz Adalid, Universitat Politècnica de València, Spain
- Gema Ibañez, Universitat Politècnica de València, Spain
- Pedro Yuste, Universitat Politècnica de València, Spain
- Daniel Gil Tomàs, Universitat Politècnica de València, Spain
- Sabina Asensio, Universitat Politècnica de València, Spain
- Beatriz Garcia-Baños, Universitat Politècnica de València, Spain
- Francisco Castells, Universitat Politècnica de València, Spain
- Diogo Nunes, University of Coimbra, Portugal
- Adriana Leal, University of Coimbra, Portugal
- Zoe Valero Ramón, Universitat Politècnica de València, Spain
- Yolanda Vives, Universitat Politècnica de València, Spain
- Javier Urchueguía, Universitat Politècnica de València, Spain

- Jose Vicente Oliver. Universitat Politècnica de València, Spain
  - Victoria Lerma, Universitat Politècnica de València, Spain
  - Edgar Lorenzo, Universitat Politècnica de València, Spain
-

---

*Table of Contents*

---

<b>Article</b>	<b>Authors</b>	<b>Pages</b>
<b>Evaluation of electrophysiological signals of <i>Pinus halepensis</i> Mill. as an indicator of forest fire risk.</b>	Rodolfo Zapata, Jose Vicente Oliver-Villanueva, Lenin G. Lemus-Zúñiga, Miguel A. Mateo Pla and Jorge E Luzuriaga	1-27
<b>Biomass assessment in homogeneous pine forest stands using Landsat images and biomass equations in the province of Palencia (Spain)</b>	David Vinué-Visús, Eloína Coll-Aliaga, José-Vicente Oliver-Villanueva, Victoria Lerma-Arce, Edgar Lorenzo-Sáez and David Fuente	28-40
<b>Development of a new AI-based methodology to assess working postures</b>	Sonia Serna Arnau, Sabina Asensio-Cuesta, Juan Carlos González García and Enrique Medina-Ripoll	41-51
<b>Comparison of the overheads provoked by the inclusion of different Error Correction Codes in Embedded Systems</b>	Joaquín Gracia-Morán, Luis-J. Saiz-Adalid, Juan C. Baraza, Daniel Gil and Pedro Gil Vicente	52-61
<b>Improving blockchain-associated energy consumption in wireless sensor networks</b>	Maythan S. Jabor, Aqeel S. Azez, José Carlos Campelo Rivadulla and Alberto Miguel Bonastre Pina	62-70
<b>Hardware Accelerating a Convolutional Neural Network Using High-Level Synthesis</b>	David De Andrés and Juan Carlos Ruiz Garcia	71-97



# Evaluation of electrophysiological signals of *Pinus halepensis* Mill. as an indicator of forest fire risk.

Rodolfo Zapata <sup>1</sup>, Jose-Vicente Oliver-Villanueva <sup>1</sup>, Lenin-Guillermo Lemus-Zúñiga <sup>1</sup>, Miguel A. Mateo Pla <sup>1</sup> and Jorge E. Luzuriaga

<sup>1</sup> ITACA - Institute of Information and Communication Technologies, Universitat Politècnica de València, València (Spain).

**Abstract.** *Pinus halepensis* forests, as Mediterranean-type ecosystems, are subject to high levels of wildfire risk in times of drought, with meteorological conditions of water stress and very high temperatures, mainly in summer. Considering the difficulty of knowing the phenological state of this species, the objective of this research was to evaluate the possibility of implementing the electrical responses as a variable in fire risk management models, compared to live fuel moisture. On the one hand, the obtained results demonstrate non-significant differences between the moisture content of the different fractions of the living branches even in times of drought with hydric stress and very high temperatures. Live fuel moisture of *Pinus halepensis* does not show significant seasonal variations under the influence of extreme fire risk factors. For this reason, it should be complemented with other variables for fire risk management models. On the other hand, the differences registered in the electrical signal show oscillations with significant variations, which are strongly correlated with the periods of extremely favourable meteorological conditions for wildfires. So, the voltages measured show ranges that correspond with great accuracy to the FWI. Voltage variation is dependent on the hydraulic dynamic plant behaviour and a result of the physiological response of pine trees to abiotic stress of drought. It is an easy-to-measure electrical parameter as well as a very reliable indicator with a high correlation with wildfire risk. Thus, electrical responses could add more knowledge about the phenological state of the trees in dependence on stress climatic conditions, allowing integration of these variables in the preventive wildfire modelling and management.

## 1 Introduction

Wildfires have been present in the Mediterranean climatic regions around the world, as a natural phenomenon long before man existed [1,2]. Wildfire is a powerful ecological and evolutionary force that regulates organismal traits, population sizes, species interactions, community composition, carbon and nutrient cycling and ecosystem functions [3,4]. Mediterranean climates are characterized by a drought season, but their length and severity can be highly variable across regions [5]. Mediterranean-type ecosystems (MTEs), with their unique climatic regime [6], support the growth of trees during the rain of spring and autumn, while the long summer drought together

with the elevated temperatures produce strong biomass desiccation [7], creating highly flammable conditions [8]. Additionally, MTEs are biodiversity hotspots located between temperate mesic climates and semi-deserts and deserts, strongly affected by climate change [9].

In all these regions, wildfires present a major disturbance to natural ecosystems, resulting in significant economic and ecological losses [4,10]. Therefore, fire risk assessment becomes a critical component of land management because it is very advantageous to anticipate the probability and magnitude of a wildfire [10]. McLauchlan et al. [4] described the diversity of ways in which fire operates as a fundamental ecological and evolutionary process and identified three emergent research challenges: the need to study fire across temporal scales, to assess the mechanisms underlying a variety of ecological feedbacks involving fire and to improve representation of fire in a range of modelling contexts. On this background, the relationship between meteorological extreme conditions and live fuel moisture is a key research topic in the complex field of fire ecology [11–21], always supported by the development of innovative technologies, especially remote sensing [22–31] and machine learning [32–34].

Forest management administrations and firefighting agencies and services are often confronted with the task of establishing proactive fire management in dependence on vulnerability and risk in different MTEs [35,36]. Many of the forecast systems are based, mainly, on meteorological data that are collected by weather stations [37], as the Canadian Forest Fire Weather Index System, which is in widespread usage globally [38]. Nevertheless, fire risk is the sum of other multiple factors [39], referring to the probability of fire ignition [37]. Globally, there is a close relationship between wildfires and anthropogenic activities, i.e., wildfires understood as human events due to negligence (e.g., agricultural burning escapes), and deliberate actions (e.g., pyromania, revenge, land use change attempts) [40], but also lightnings are more and more frequent natural causes [41], especially in large unmanaged forestlands [42].

Three major influencing factors with their respective variables intervene in the behaviour of an extreme wildfire: meteorology (wind regime, temperature, and relative humidity), topography (terrain configuration in terms of slope, accessibility, and vegetative structure) and the state of the forest fuel (moisture and flammability characteristics) [43]. All factors are closely related and jointly influence the vulnerability of forest fuel to igniting, but several authors point out the mechanisms through which plant responses to drought and, consequently, to forest flammability, specifically live fuel moisture, but also physiological reactions to water stress in the ecosystem (soil water content and plant traits, including rooting patterns and leaf traits such as the turgor loss point, osmotic potential, sap flow, elasticity and leaf mass ratio of dead to live fuels, etc.) [44,45].

In fact, the relationships between drought and wildfires are well documented for MTEs, especially in Europe [46,47], Australia [48,49] and North America [50], with wild-fire occurrence and area clearly increasing in response to drought [51,52]. There is also evidence that drought interacts with other controls (wind regimes, topography, forest management activities) to affect fire intensity, severity, extent, and frequency [50,53]. Due to drought directly influences vegetation dryness in trees and shrubs [21], live fuel moisture has been identified as a key factor of significance in biomass combustibility [54–58]. In this context, it should be considered that the large plant diversity that characterizes MTEs is associated with the success of coexisting species

in avoiding competition for soil resources (water and nutrients) by differential exploitation in space (soil layers) and light time (year and daily) [59]. So, live fuel moisture content is influenced by meteorological and soil variables, but mainly by the plant species and its phenological state, and hydraulic behaviour [58]. Therefore, given this influence, it is considered necessary to expand the knowledge regarding the phenological state of the different species present in the ecosystem, to know its magnitude and its seasonal variation, and to understand and predict fire behaviour [58], by testing several innovative technologies of terrestrial [60,61] and remote sensing [62].

Rothermel [63] proposed a classification of the humidity of live fuel; this classification was based on the stage of development of the vegetation. However, the data obtained in studies carried out in some MTEs differ from this classification [19,64,65], since they find a differentiated behaviour of live fuel moisture according to each species and strongly depending on seasonality [23,66]. Several authors have studied the phenological state of the vegetation by directly measuring the live fuel moisture content by taking physical samples in the field [58,67]. Through these works, it was able to relate the phenological state of the plants with their fuel moisture content, at least in a large part of the species that inhabit the MTEs [58]. Nevertheless, for one of the most important tree species in the Mediterranean basin, such as *Pinus halepensis* [51,68–72], the results on the variation in seasonal moisture were not conclusive [58]. Since *Pinus halepensis* hardly show variation in moisture content throughout the vegetative cycle; neither show moisture variations in the face of extreme heat and drought episodes that are usually recorded in the Western European Mediterranean area [58].

Moreover, for a better understanding the phenological state of the plants in MTEs in dependence on severe drought conditions, different plant hydraulic traits have been analysed, such as the measurement of the sap flow, which was proposed for the evaluation of transpiration rates [73]. However, some authors claim that sap flow measurements only provide information on the water movement within plants and are not directly related to the rapid responses to environment or climatic stress [74]. Other authors have analysed other plant hydraulic traits, such as saturated moisture content, cell wall rigidity or turgor loss point, cell solute potential, symplastic water fraction and tissue capacitance [75].

Furthermore, some authors have been proposed the measurement of electric potential as a valid method to evaluate the phenological state and stress responses of trees and shrubs [76]. The existence of a continuous electric potential between the electrodes inserted in the tree phloem and the surrounding soil was discovered and described many years ago [77–80]. This electric potential is associated with electrochemical effects that include membrane diffusion potentials and active transport of ions [81]. In addition, more recent works documented that some environmental stimuli also produce changes in electrical signals of trees [82]. More specifically on *Pinus halepensis*, a high correlation between meteorological variables and variations in electrical signals could be demonstrated [61]. According to some authors [83–85], electrical properties (as plant physiological reaction) seem to be related to rapid responses to water stress. Unfortunately, these responses are sometimes not evident or do not produce consistent visual indicators, such as wilting and changes in leaf colour [86].

Considering these characteristics of electrical signal measured on trees, the increasing importance of *Pinus halepensis* in European Mediterranean forests [51,58,68–72], and the difficulty of knowing the phenological state of this species [58], the objective of this re-search was to preliminarily evaluate the possibility of implementing the electrical signals responses of *Pinus halepensis* as an indicator to complement live fuel moisture assessment, as a method to monitor the phenological state and the drought stress level. Finally, the re-search aims to analyse the relationship between the obtained results for electrical signals and the wildfire risk rating, normally used by firefighting agencies.

## 2 Material and Methods

### 2.1 Research Design

This main objective of the research is to study the relationship between the electrical signals as a result of phenological state of *Pinus halepensis*, live fuel moisture content and wildfire risk. Therefore, the main parameters to be measured will be the components of the electrical signal: voltage and short-circuit current [87], together with measurements of the moisture content of the live fuel, as well as also with the main meteorological conditions (temperature, relative humidity, and rainfall) and wildfire risk index published by local meteorological or emergency agencies.

To do this, the research has been divided in two phases:

1. In a first stage, the measurements were carried out during the main wildfire season (24 weeks in the hottest and driest months, from end of spring to beginning of autumn) in a representative area of the Mediterranean basin.
2. The results of this first phase were complemented with a second survey, in which the values of the electrical signals collected in previous three years were retrospectively analysed.

### 2.2 Selection of Sample Stand

*Pinus halepensis* Mill. (Aleppo pine) and *Pinus brutia* Ten. (Turkish or Calabrian pine) are two systematically close tree species dominating MTEs in the European basin [24], which can naturally hybridize where they co-occur [88]. So, both are drought-tolerant and fast growing native coniferous species [24], well adapted to dry summer conditions [70]. *Pinus halepensis* widely covers the western side of the basin, while *Pinus brutia* is located mainly on the eastern side, both mainly at coastal zones [89]. They are among the species most affected by wildfires in Europe [90], although

they are fire resilient trees due to the high production of serotinous cones that favour a quick post-fire regeneration [91]. These species have been widely planted between 1930 and 1980 in Mediterranean areas for soil protection and windbreaks near the coasts [92]. Specifically, *Pinus halepensis* is the most widely distributed and abundant pine in MTEs, covering nearly 7 million ha in this region [93], being present in all regions on both shores of the Mediterranean Sea and extending from the Western Mediterranean (Spain, Morocco), where it is most abundant, to Lebanon through Southern France, Italy, Greece and Turkey in South Europe and Algeria, Tunisia, Libya in North Africa. Bioclimatic envelope models predict that the suitable climatic area of *Pinus halepensis* is clearly in expansion [94–96]. Thus, we have selected *Pinus Halepensis* because of its increasing importance and representativeness in the MTEs [97]. Actually, it is the most influential tree species in the total availability of biofuel in the Mediterranean forests [58]. Furthermore, Soriano [58] describes that *Pinus halepensis* is not showing strong variations in its moisture content in the face of seasonal variations, so that possible variations of electrical responses could add more knowledge about the phenological state of the trees in dependence on severe drought conditions, allowing for the integration of these variables in the preventive wildfire management.

As in previously published works [61,98], we decided to use as a study area a representative young forest composed of 93% of *Pinus halepensis* from a post-fire natural regeneration, located within the protected area of the Sierra Calderona Natural Park in Gátova, Valencia (Spain). The selected stand is located at latitude 39°45'28.80" N and longitude 0°30'36.36" W. The forest stand has a population density of 484 trees/ha, with an average DBH of 12.10 cm, a tree height of 5.16 m, and an age of 27 years (Figure 1).



**Figure 1.** View of the sample stand (August 2021).

This sample stand was selected because it is an even-aged forest with enough homogeneous trees, within an area not affected by significant natural disturbances, such as re-cent wildfires, pests, or damage due to hurricanes or heavy snowfall. It was sought that the stand had these conditions of even-age (natural regeneration after a previous wildfire occurred in summer of year 1994), to reduce the differences between the individuals that are part of the study since, as we have seen in previous works, age is an influencing factor in the amplitude of the electrical signal [98]. The selected stand meets all the characteristics above described, with constant site conditions (soil and water, orientation, and slope).

A soil analysis based on edaphic profiles was carried out to obtain the main soil variables that can influence on the electrical behaviour of the sample trees (Table 1).

**Table 1.** Soil properties in pilot stand.

Texture (%)						
Sand	Silt	Clay	USDA Classification			
20.1	43.6	12.3	Loam			
Moisture Factor	Ph	Electric Conductivity [ds/cm]	Wp [%]	Field Capacity [%]	Pore Space [%]	Depth [cm]
0.92	8.2	1.869	3.7	11.8	45.66	30

### 2.3 Selection of Sample Trees

The first phase of this study was carried out on a total of 240 trees, selected following the method described by Hapla and Saborowski [99] and used by other authors [100,101] for sampling representative trees in a forest stand for analysing physical wood characteristics.

Although the electrical measurement process is non-destructive (see Section 2.4.1 (a)), obtaining the moisture content (see Section 2.4.1 (b)) requires cutting plant material (living branches) from the standing trees in different testing times. This is obviously not possible without seriously injuring the trees. To avoid this impact, the total set of 240 trees has been subdivided into 24 groups, i.e., one group of 10 trees per evaluation week. Each weekly sample group was formed by the 10 trees closest to the centre placed ground electrode.

In the second phase of this work, the electrical signal values obtained in previous works [61,98] were measured in fifteen representative trees of the same stand over three years. These trees were also selected using the representative tree selection method de-scribed [61,98,99].

## 2.4. Measurement Procedures

In each of the 24 weeks of the first phase, the values of the two components of the electrical signal (voltage  $V$  and short-circuit current  $ISC$ ) were recorded and the moisture content of ten trees was measured each week.

These measurements were carried out in the 24 hottest and driest weeks of the year, from one month before the start of the official wildfire season in the Region of Valencia/Spain (May 2021) until one month after the end of it (October 2021). All weekly measurements were taken on Saturdays at 12:00 pm CET, since this is the time for which wild-fire risk predictions are made according to the Fire Weather Index (FWI) system of the Spanish Meteorological Agency (AEMET) [102]. In addition to that, as could be demonstrated in previous works, the central hours of the day, close to the zenith, are the moments in which the tree presents the minimum value of electrical signal [61].

For the second phase, the voltage ( $V$ ) and short-circuit current ( $ISC$ ) values used in our previous work [61], were retrospectively evaluated. These electric signal values were collected weekly, using the same data acquisition protocol as used in the present study, since May 2018.

### 2.4.1. Measurement of Electrical Signal

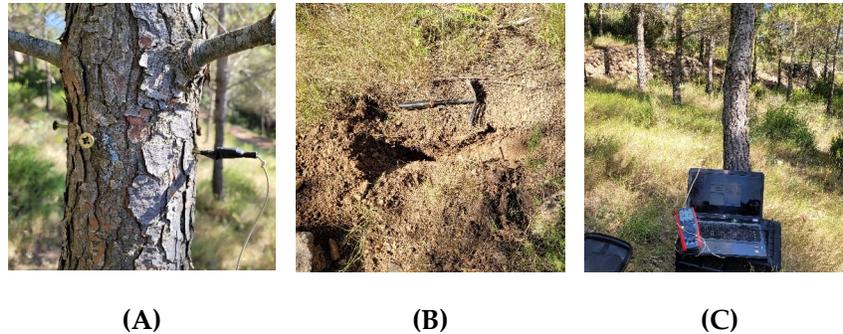
The values of voltage ( $V$ ) and short-circuit current ( $ISC$ ) of the trees were measured between the electrodes inserted in the trunk and the electrode buried in the ground. To carry out this work, electrodes with the same characteristics as those of our previous works [61,98] as well as those used by other authors [61,76,81,98,103,104] were used.

Two electrode types were used according to their function.

(a) The electrode used in the tree, made of stainless steel, was inserted directly into the trunk at 1.5 m above the ground at a depth sufficient to ensure contact with the phloem tissue. A screw shape was chosen due to its greater ease of insertion and removal from the trees, causing only a minor wound. In addition, thanks to the screw spiral, these electrodes have a larger contact surface with the vegetative tissue compared to smooth cylindrical electrodes. So, these electrodes were inserted into the trunk ensuring contact with the phloem tissue, by inserting them with a torque wrench, which allowed us to detect the change in tissue hardness. This last action we consider fundamental, since electrical signals are more easily transmitted throughout this tissue, given its lower resistance to electrical flow, compared to other plant tissues [82].

(b) The second type was a non-polarized platinum electrode [105], which was used as ground reference. These electrodes were buried in the mineral soil at a depth of between 20–25 cm once the top layer of topsoil had been removed. It should be noted that given the natural conditions in which the experiments were carried out, we could not install the reference electrode in a greater depth due to the soil hardness and the presence of rocks.

Both electrodes were connected to the measuring equipment through electrical connectors and a 0.5 mm copper conductor cable insulated with a flexible plastic sheath (CE 0123) (see Figure 2).



**Figure 2.** Tree electrode (A), ground electrode placement (B), arrangement of measuring equipment (C).

The equipment used for the voltage (V) and short-circuit current (ISC) measurement was a multimeter UT71D UNIT with an input impedance of 2.5 G $\Omega$  and a precision of 0.1%  $\pm$  2 mV.

#### 2.4.2. Measurement of Moisture Content

Physical samples were taken to the laboratory, following the methodological recommendations of the National Forest Research Institute of Spain [106,107].

Field work:

The following samples were taken from the first live and healthy branch from the bottom of the tree-crown:

Fraction 1 (BB): samples were taken from the base of the branch on a weekly frequency, with diameters of 20–30 mm and 5–10 cm length, without needles, in order to compare it with the non-destructive moisture content methodology.

Fraction 2 (BM): samples were taken from the middle of the branch on a monthly frequency, with diameters of 10–20 mm and 5–10 cm length, without needles.

Fraction 3 (LF): samples were taken from end part of the branch on a monthly frequency, with diameters <10 mm, with twigs and needles, without cones.

Samples were taken always on Saturdays between 12:05 and 2:00 pm CEST. Each sample was placed in a hermetically sealed plastic container, identified with the reference data and transported immediately to the laboratory.

Laboratory work:

The samples were weighed on a precision balance in the green state. After being dried in an oven at 105 °C for 24 h until constant weight was obtained, they were weighed in anhydrous state. The moisture content (MC%) is calculated with the formula:

$$MC\% = \frac{\text{wet weight} - \text{weight after drying}}{\text{weight after drying}} * 100$$

## 2.5. Meteorological Time-Series

The meteorological data for the area was provided by a professional meteorological station installed at 39°46'10.12" N, 00°31'14.19" W. The meteorological station is a Davis Vantage VUE model owned by the Valencian Meteorological Association (AVAMET) [108].

## 2.6. Wildfire Risk Assessment

The Spanish State Meteorological Agency (AEMET) publishes the official FWI daily [104]. The data is open and available online through its web services. We have used the data recorded to the region of Valencia, defined as Zone 3, where our pilot stand is located. The FWI levels are coloured from green (low risk) to red (extreme risk).

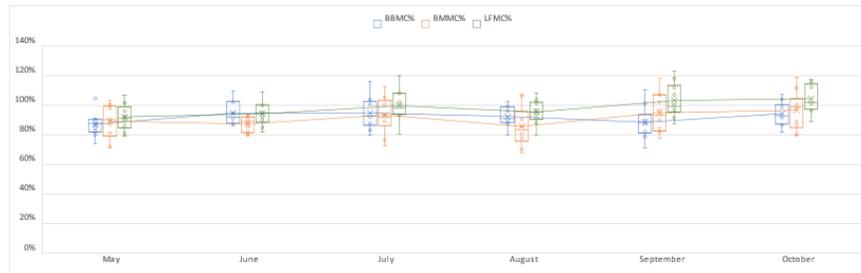
**Table 2.** Results of the analysis of variance (ANOVA) among BBMC%, BMMC% and LPMC%.

Sample	F	Pr > F	p-Value	$\alpha$	Result
May	0.714	0.499	0.381	0.05	non-significant differences
June	1.278	0.295	0.837	0.05	non-significant differences
July	1.211	0.314	0.863	0.05	non-significant differences
August	1.461	0.250	0.295	0.05	non-significant differences
September	3.793	0.035	0.646	0.05	non-significant differences
October	2.516	0.100	0.389	0.05	non-significant differences
Total Period	7.367	0.001	0.208	0.05	non-significant differences

### 3. Results and Discussions

#### 3.1. Comparison between Moisture Content in Different Live Branch Fractions

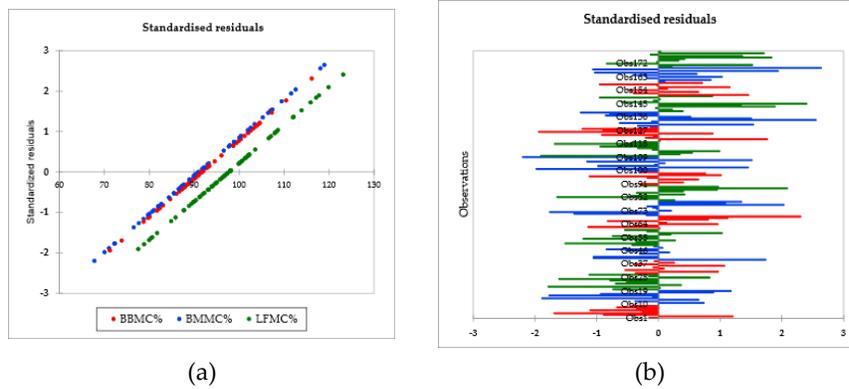
Figure 3 shows the obtained laboratory data for the moisture content in the three branch fractions during the six months of measurement. For all measurements taken during this period, the mean value obtained for the moisture content measured at the branch base (BBMC%) is 91.92%, with a standard deviation of  $\text{std } \pm 9.12\%$ , very similar to the mean value for the moisture content measured at the middle of the branch (BMMC% = 91.18%,  $\text{std } \pm 12.06\%$ ) and slightly under the mean value obtained for the fine live fuel moisture content (LFMC% = 98.17%,  $\text{std } \pm 10.27\%$ ). Other authors reported very similar LFMC% measured in needles and twigs for Mediterranean pines [18,58,109].



**Figure 3.** Comparison between moisture content in different branch fractions of *Pinus halepensis* from May to October 2021: branch base (BBMC%, middle of the branch (BMMC%) and fine live fuel (LFMC%).

In order to analyse possible significance differences among BBMC%, BMMC% and FMC%, several ANOVA tests are performed for the total sampling period and for each month among the three fractions. Table 2 shows the results.

Although the values obtained for LFMC% are slightly higher than BBMC% and BMMC% for the total sample, as well as for the individual months (between 5 and 10%), the variance analysis demonstrate that these differences are non-significant, neither for the individual samples in each month nor for the total sample. The standardized residual analyses shown in Figure 4 also demonstrate this result.

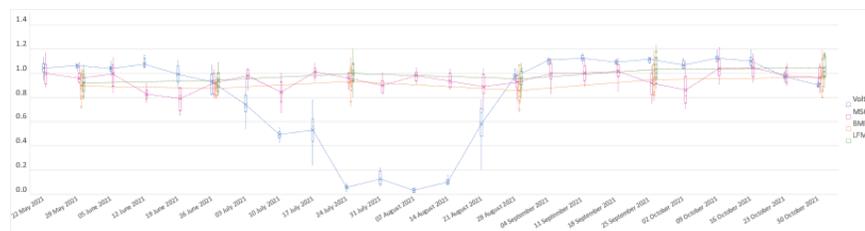


**Figure 4.** Standardised residual analyses among BBMC%, BMMC% and LFMC% for the total period (a) and standardised residual vs total observations (b)

The results obtained demonstrate that there are non-significant differences among the moisture content among the three branch fractions (BBMC%, BMMC% and LFMC%). In this sense, samples of live branches bases can be representative for the moisture content for entire live fuel. Following Mitsopoulos and Dimitrakopoulos [110], the live aerial fuels that are consumed during crown fires in *Pinus halepensis* forests are composed of needles (16.7%), twigs with 0.0–0.63 cm diameter (12.6%), branches with 0.64–7.5-cm diameter (62.7%), and branches >7.5-cm diameter (3.7%). Taking BBMC% as a reference measurement at the base of the first living branch in standing trees opens an opportunity to monitoring through sensorised moisture meters, e.g., by electrical resistance or capacitive devices.

### 3.2. Seasonal Variability of Live Fuel Moisture Content

Figure 5 visually shows the values obtained for the moisture content of the three fractions (BBMC%, BMMC% and LFMC%) of the first living branch during the 24 weeks measurement in the critical wildfire risk season, i.e., from late spring to early autumn 2021.



**Figure 5.** Seasonal variability of moisture content of the first living branch (BBMC%, BMMC% and LFMC%) in comparison with voltage during the 24 weeks from late spring to early autumn 2021.

The first visual analysis indicates that *Pinus halepensis* does not present clear seasonal variations of moisture content, neither in the woody fractions of the branches (BBMC% and BMMC%) nor in the fine live material (LFMC%), even during the water stress conditions of the hot and dry summer weeks. An ANOVA test among the 24 mean values of the weekly measurements for BBMC% shows also non-significant differences ( $F = 1.38$ ,  $p < 0.001$ ). Furthermore, non-significant differences can be demonstrated among the monthly measurements of BBMC% ( $F = 1.34$ ,  $p < 0.001$ ) as well as LFMC% ( $F = 2.75$ ,  $p < 0.001$ ). Finally, Figure 5 also shows through the Box and Whiskers plots that the variation among the 10 measured trees in each measurement trial is very low. Additional individual ANOVA tests demonstrate the non-significant differences of moisture content for the three variables (BBMC%, BMMC% and FLMC%) among trees ( $p < 0.001$ ). Other authors reported very similar behaviour of LFMC% of Mediterranean pines [18,58,109], also not being able to demonstrate representative and significance seasonal variations.

### 3.3. Seasonal Variability of Electrical Signals in the Trees

Table 3 present the total results obtained for the electric signal measurements: voltage (V) and short-circuit current (ISC). Both V and ISC present much higher variations than the moisture measurements. So, V has an average value of  $V = 0.808$  V with very high standard deviation ( $\text{std} = \pm 0.381$  V) and variance. The same heterogeneous behaviour present ISC with an average value of  $\text{ISC} = 1.998$  mA and  $\text{std} = \pm 2.531$  mA.

**Table 3.** Descriptive statistics for the results of voltage (V) and short-circuit current (ISC) from 240 samples measured in 24 weeks from late spring to early autumn 2021.

	Electrical Signals	
	Voltage (V) [V]	Short-Circuit Current (ISC) [ $\mu\text{A}$ ]
Mean	0.808	1.998
Median	0.990	1.100
Minimum	0.032	0.000
Maximum	1.124	15.690
Standard deviation (n - 1)	0.381	2.531
1st Quartile	0.569	0.238
3rd Quartile	1.081	2.735
Variance (n - 1)	0.145	6.408

These heterogeneous values can be explained by the analysis of the seasonal variations. So, Figure 5 also shows the results obtained for the voltage (V) measured in the trees during the 24 weeks of higher wildfire risk from late spring to early autumn 2021.

The results obtained clearly show that the electrical signals measured in the trees coincide with the two growth seasons of *Pinus halepensis*, specifically spring and autumn [111,112]. Therefore, with this, we can associate the moments in which the environmental conditions are favourable for the growth of the trees with the highest values of the electrical signal. On the contrary, the periods in which the environmental conditions become more difficult for the survival of the pines, specifically during heavy drought conditions of continuous lack of rain and very high temperatures, the tree gives us a clear reduction in the electrical signal, which is observed more clearly in the case of voltage (V), with reductions of up to 90% and, in any case, of more than 50%.

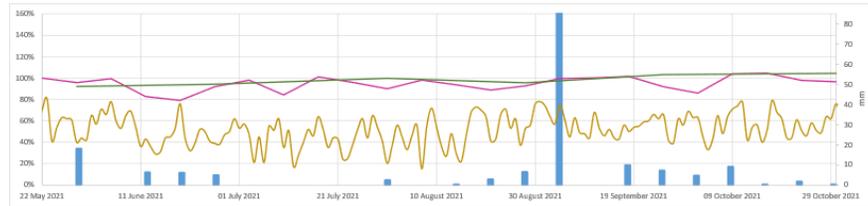
Hence, the seasonal variations of voltage in *Pinus halepensis* observed can be interpreted as a direct result of a loss of conductivity during the strong climatic conditions in the driest summer days. In this sense, Fotelli et al. [113] demonstrate that during xerothermic periods, *Pinus halepensis* has typical isohydric behaviour: maximum photosynthesis, sap flow and stomatal conductance declined through stomatal control to limit water potential reduction and loss of conductivity. This loss of sap flow and conductivity has been also observed in several studies [114]. Electrical responses such as voltage seem to be directly related with sap flow conductivity, so that further research activity to analyse in detail this relationship should be carried out.

### 3.4. Relationship between Life Fuel Moisture and Electrical Signals

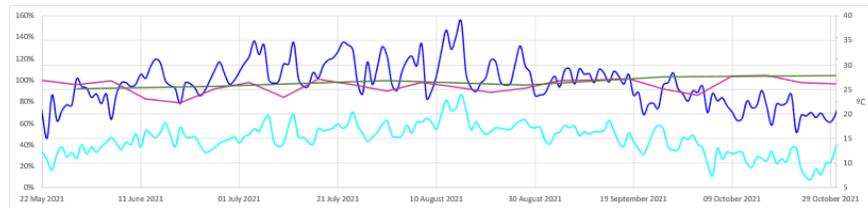
As shown in the previous chapters, the live fuel moisture content of *Pinus halepensis* do not present significant variations throughout the warmest and driest part of the year, without showing a significant decrease in the hot summer months, on the contrary that the electrical signals do, especially the voltage. So, the detailed analysis shows that the voltage curves remain more or less constant during the months of May and June, as well as from the beginning of September to the end of October. However, as soon as extreme summer conditions with high temperatures and very low rainfall dominate the central summer months (July and August), the voltage values drop very clearly and significantly. The more or less constant voltage of around 1 V decreases clearly under <0.5 V under the nine weeks between beginning July and End of August, and even dramatically during the central weeks of end of July and beginning of August with <0.1 V. In fact, the moisture content and voltage curves are closely aligned during May and June, decoupling very clearly during the central part of the summer in July and August, and re-coupling from the beginning of September.

Moreover, Figures 6 and 7 show that precipitation, temperature, and relative air humidity do not exert a direct influence on live fuel moisture content of *Pinus halepensis*. Not even noticeable changes in BBMC%, BMMC% or LFMC% can be observed on the central summer weeks of end of July and beginning of August, with the hottest and driest days, especially the 24 July and 7 August 2021, which were

classified as an extreme risk by the weather and firefighting agencies, following the FWI criteria [102].



**Figure 6.** Moisture content (BBMC%, purple line and LFMC%, green line) vs. weekly-accumulated precipitation (blue column) and relative humidity (light brown line) during the 24 weeks study.



**Figure 7.** Moisture content (BBMC%, purple line and LFMC%, green line) vs. maximum (dark blue line) and minimum (light blue line) daily temperatures.

The results obtained demonstrate that the moisture content (BBMC% or even LFMC%) do not present representative seasonal variations in *Pinus halepensis* and is not able to react directly or quickly to extreme drought conditions. Characterization of the fuel structure and its relevance for fire behaviour has been the topic of much research in MTEs [115]. Thus, variations in LFMC% are often taken into account, although some discussions are still active on its role in fire propagation [116]. Some studies have addressed the role of FMC on fire behaviour [117]. Others have addressed how canopy drying, following bark beetle attacks, for instance, impacts fire behaviour [118–120].

In our research, the moisture content measured at the living branches of *Pinus halepensis* (including LFMC%) shows no representative seasonal variations, with values between 80 and 100%, which is consistent with other studies with this species [109]. Although LFMC% is considered as an important determinant of forest flammability under Mediterranean conditions [45], other authors demonstrate that even other Mediterranean pines show also very limited LFMC% seasonal variations: *Pinus pinaster* 90%–100% [121] and *Pinus nigra*: 95%–115% [122]. These pines exhibit isohydric behaviour [113], i.e., little variation in midday leaf water potential, and relatively tight regulation of stomata in response to soil drying [122]. Mediterranean

pinus are adapted to the xerothermic conditions (high temperatures and droughts) of summertime, due to its drought avoidance strategy of reducing stomatal conductance under water shortage [123]. *Pinus halepensis* displayed a water-saving, drought avoidance (isohydric) strategy via stomatal control in response to summer drought [124,125]. The species benefited from periods of high available soil water (normally autumn to spring) [126,127]. These drought episodes do not influence directly moisture content at branches and leaf level [58,128], so that direct measurement of LFMC% of Mediterranean pines should not be considered as the only appropriate indicator to monitor wildfire risk. Moreover, strong drought conditions and consequently high evapotranspiration carries physiological responses in branches and needles of *Pinus halepensis*. Thus, Fotelli et al. [113] demonstrate that during xerothermic periods, typical isohydric behaviour was exhibited by *Pinus halepensis*: maximum photosynthesis, sap flow and stomatal conductance declined through stomatal control to limit water potential reduction and loss of conductivity. This loss of sap flow and conductivity has been also observed in several studies [114]. However, in periods when water availability was not a limiting factor, this species was able to maximize its carbon gain if other controlling parameters, such as air temperature and net radiation, simultaneously ensured a favourable environmental regime [113].

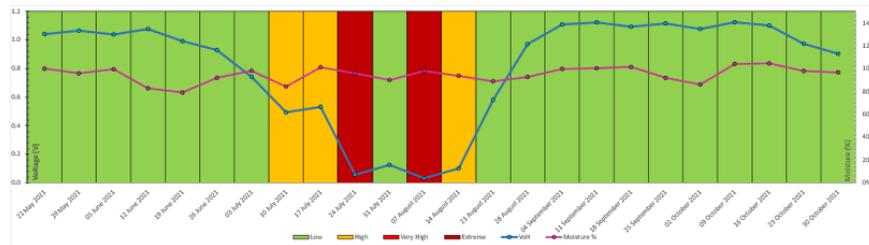
On the contrary, LFMC% does present representative seasonal variations in some Mediterranean shrub species. Thus, compared to the results obtained for *Pinus halepensis* (and those observed for other Mediterranean pines), the main shrubs that are part of MTEs behave significantly differently [129]. Especially relevant is the seasonal variation in LFMC% of *Rosmarinus officinalis*, with minimum values of 40% in summer and maximum values of 140% in autumn, winter, and spring [19,109,112]. Also, other species, e.g., *Ulex parviflorus* (60%...120%), *Erica multiflora* (50%...90%) or *Juniperus oxycedrus* (65%...100%) shows very clear seasonal variations [58,111]. This undoubtedly explains the importance that the summer drop in the LFMC% has on the vulnerability of wildfire in shrublands [130]. Undoubtedly, for future research, it would be very interesting to relate the variation in the LFMC% and its relationship with the voltage in these shrubs, as they have shown very similar behaviour patterns.

Following our results, it seems to be very difficult to assess and monitor vulnerability of Mediterranean pine forests to wildfire risk only in dependence on LFMC% of pine trees. Additionally, other physiological plant traits seem to better explain the high wildfire risk in times of drought, especially osmotic potential, sap flow, wilting and needle senescence or dead fuel presence and evolution. Fuel dynamics, as a result of the physiological response of the pine trees to drought conditions, have to be analysed integrating variables resulting from water stress (soil water content and plant traits, including rooting patterns, and leaf traits such as the turgor loss point, osmotic potential, sap flow, elasticity and leaf mass per area), but also the ratio of dead to live fuels [131,132]. In this sense, needle cavitation and subsequent shedding is of particular relevance for pines, transforming green live fuel into dead fuel, which are totally dry, and thus easier to ignite [45]. Therefore, it is necessary to investigate the seasonal fuel dynamics from a more integral perspective, complementing the measurement of LFMC%. All variables resulting from physiological responses that can influence fire risk and that can be easily measured and monitored, including electrical signals, should be integrated into the risk models.

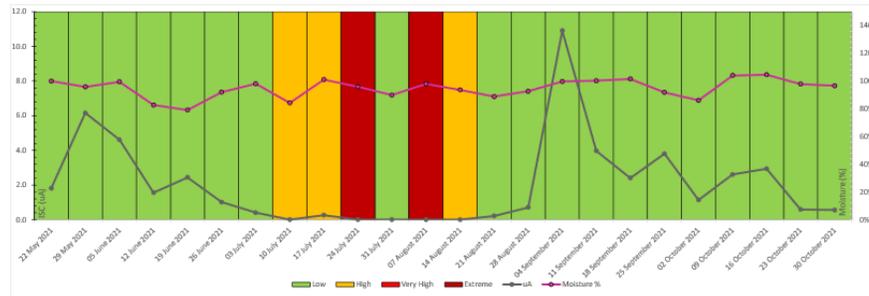
### 3.5. Relationship between Electric Signals and Wildfire Risk

(a) Assessment for the 24 weeks study in year 2021

A first analysis of the relationship between the electrical signals and the wildfire risk has been carried out, by comparing the mean values of both V and the ISC for each week (always measured on Saturdays) with the wildfire risk following the FWI criteria of those same days, considering for this daily maximum temperature, minimum relative humidity, maximum wind speed and rainfall. The results obtained are shown graphically in Figures 8 and 9, as well in the supplementary material.



**Figure 8.** Mean values of V (blue line) and BBMC% (pink line) vs. FWI classification of the measurements according to FWI criteria during the 24 weeks study (background colour).



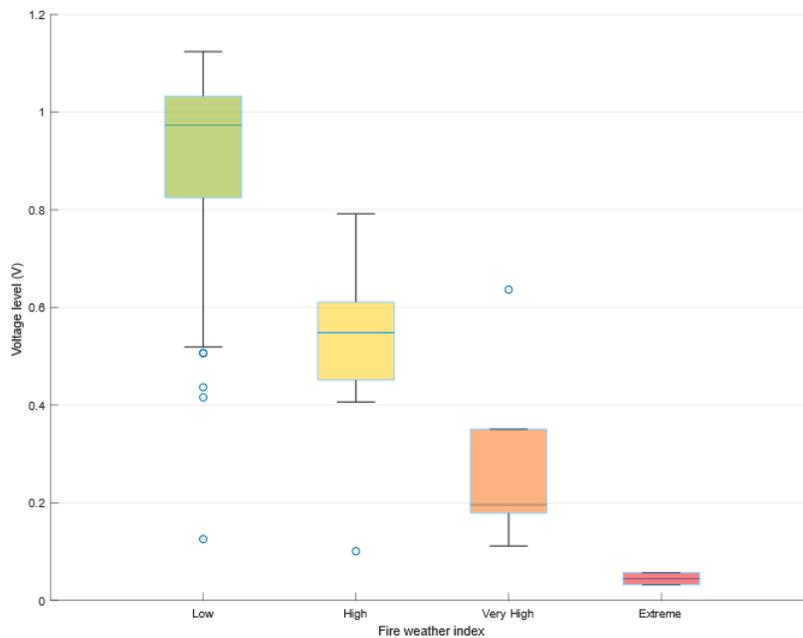
**Figure 9.** Mean values of ISC (tagged uA in the figure with a dark grey line) and BBMC% (pink line) vs. FWI classification of the measurements according to FWI criteria during the 24 weeks study (back-ground colour).

The results obtained cannot demonstrate a representative correlation between moisture content and FWI. The highest FWI values, and consequently the brunt of the fire season in the Western Mediterranean Basin occurs normally during July [132], while the observed moisture content values do not show any reduction, as also observed by Qi et al. [131]. Soler Martin et al. [121] demonstrate that no seasonal changes of LFMC were recorded during summer in needles and small branches in *Pinus pinaster* stands, contrary to predictions from the FWI, which fully matches with the results obtained in our study for *Pinus halepensis*.

These results demonstrate that both electrical signals measured (V and ISC) show a noticeable reduction during the summer period, reaching the minimum values on the days in which the FWI in the pilot stand was classified as extreme [102]. However, the results for V are much clearer and more significant related to FWI than the results obtained for ISC.

(b) Assessment for three years survey (2018–2021)

Measurements for electrical signals, specifically voltage, have been performed in the pilot stand since May 2018 until October 2021, so that we can make a long-term evaluation of the relationship between V and FWI. Figure 10 shows in Box and Whiskers plots the average, standard deviation and minimum-maximum for all days during the three years classified by wildfire risk categories (low, high, very high, extreme) following FWI criteria.



**Figure 10.** Voltage values for all measured days from May 2018 to October 2021 classified by FWI categories (low, high, very high and extreme wildfire risk).

The first interpretation of the results allows us to observe that the mean values obtained for V in the four FWI categories are significantly different, clearly reducing the voltage as the risk increases. An ANOVA test also corroborates the observed differences as statistically highly significant ( $F = 39.138$ ,  $p < 0.001$ ). Thus, while FWI low

presents an average of 0.90 V, the FWI high class decreases to 0.53 V and the FWI very high class to 0.28 V, with the FWI extreme class reaching a mean of only 0.04 V.

Finally, this analysis also shows that on days with low fire risk (FWI low) practically all the values (96%) greatly exceed  $>0.5$  V. Even 81% exceed the threshold of  $>0.8$  V. On the other side, on the days classified as very high risk (FWI very high) and extreme risk (FWI extreme), only one of all values (88%) exceed  $>0.4$  V, even 100% of the extreme risk does not exceeds even  $>0.1$  V.

The voltage level is a result of the physiological response of *Pinus halepensis* to the abiotic stress of drought in summer. It is an easy-to-measure electrical parameter as well as a very reliable indicator with a high correlation [133] with wildfire risk. Having obtained a Spearman's Rank correlation coefficient of 0.6816 (p-value  $< 0.001$ ) between the FWI index and the raw voltage values for the 24-week study in 2021, the same test for the period 2018–2021 increases to 0.7816 (p-value  $< 0.001$ ). We have to notice, that our research is a pioneer study to link electrical signals with plant physiology in a context of wildfire risk management, and our findings demonstrate the potential of incorporating electrical responses as one of the ecophysiological plant traits to investigating seasonal changes in wildfire ignition risk and flammability.

## 4. Conclusions

The most important conclusions that we can draw from the research are the following:

- No significant differences have been observed between the moisture content of the different fractions of the branches of *Pinus halepensis* (base of the branch, half of the branch and twigs and needles as live fuel), even in times of drought with hydric stress and very high temperatures.
- Live fuel moisture content has not shown significant variations under the influence of extreme fire risk factors in the summer time. For this reason, it should be complemented by other reliable variables for fire risk assessment and monitoring in MTEs dominated by *Pinus halepensis*. Thus, other plant physiological traits have to be integrated in the assessment and modelling of the high risk of wildfires in *Pinus halepensis* stands in times of water stress and high temperatures, related both to hydraulic dynamics (osmotic potential, sap flow) and dead fuel (wilting and needle senescence, dead fuel presence and evolution). However, as LFMC% responds better to fire risk conditions in some shrub species in MTEs, we propose to analyse in-depth the relationship between LFMC% and electrical responses in these shrubs.
- The variations registered in the electrical signal generated in *Pinus halepensis* show oscillations with significant variations, which are strongly correlated with the periods of extremely favourable meteorological conditions for wildfires (Spearman rho of 0.78).
- The voltages measured show ranges that correspond with great accuracy to the official fire risk levels based on the FWI system.
- The electrical signals, specifically voltage, are a result of the physiological response of the Mediterranean pine trees to the abiotic stress of drought in summer. It is an easy-to-measure electrical parameter as well as a very reliable indicator with a high correlation with wildfire risk.
- Electrical responses could add more knowledge about the phenological state of the trees in dependence on stress climatic conditions, allowing for the integration of these variables in the preventive wildfire management. Although for this we also consider that a more in-depth investigation is necessary.
- Finally, the results obtained and the knowledge gained allows for the exploration of new possibilities for the development of wireless terrestrial sensors based on voltage measurement, which allow online monitoring of the risk of wildfire ignition and propagation with potentially maximum spatial and temporal resolution.

## References

1. Naveh, Z. The evolutionary significance of fire in the Mediterranean region. *Vegetatio* **1975**, *29*, 199–208.
2. Bodí, M.B.; Muñoz-Santa, I.; Armero, C.; Doerr, S.H.; Mataix-Solera, J.; Cerdà, A. Spatial and temporal variations of water repellency and probability of its occurrence in calcareous Mediterranean rangeland soils affected by fires. *Catena* **2013**, *108*, 14–25.
3. Pausas, G.J.; Keeley, J.E. A burning story: The role of fire in the history of life. *BioScience* **2009**, *59*, 593–601.
4. McLauchlan, K.K.; Higuera, P.E.; Miesel, J.; Rogers, B.M.; Schweitzer, J.; Shuman, J.K.; Tepley, A.J.; Varner, J.M.; Veblen, T.T.; Adalsteinsson, S.A.; et al. Fire as a fundamental ecological process: Research advances and frontiers. *J. Ecol.* **2020**, *108*, 2047–2069.
5. Hoerling, M.; Eischeid, J.; Perlwitz, J.; Quan, X.; Zhang, T.; Pegion, P. On the increased frequency of Mediterranean drought. *J. Clim.* **2012**, *25*, 2146–2161. <https://doi.org/10.1175/JCLI-D-11-00296.1>.
6. Rundel, P.W. Landscape disturbance in Mediterranean-type ecosystems: An overview. In *Landscape Disturbance and Biodiversity in Mediterranean-Type Ecosystems*; Springer-Verlag: Berlin/Heidelberg, Germany, 1998; pp. 3–22. [https://doi.org/10.1007/978-3-662-03543-6\\_1](https://doi.org/10.1007/978-3-662-03543-6_1).
7. Kelly, A.E.; Goulden, M.L. A montane Mediterranean climate supports year-round photosynthesis and high forest biomass. *Tree Physiol.* **2016**, *36*, 459–468. <https://doi.org/10.1093/treephys/tpv131>.
8. Montenegro, G.; Ginocchio, R.; Segura, A.; Keely, J.E.; Gómez, M. Regímenes de incendios y respuestas de la vegetación en dos regiones de clima mediterráneo. *Rev. Chil. De Hist. Nat.* **2004**, *77*, 455–464.
9. Ramirez-Valiente, J.A.; del Blanco, L.S.; Alía, R.; Robledo-Arnuncio, J.J.; Climent, J. Adaptation of Mediterranean forest species to climate: Lessons from common garden experiments. *J. Ecol.* **2022**, *110*, 1022–1042. <https://doi.org/10.1111/1365-2745.13730>.
10. Vasilakos, C.; Kalabokidis, K.; Hatzopoulos, J.; Kallos, G.; Matsinos, Y. Integrating new methods and tools in fire danger rating. *Int. J. Wildland Fire* **2007**, *16*, 306–316.
11. Boer, M.M.; Nolan, R.H.; De Dios, V.R.; Clarke, H.; Price, O.F.; Bradstock, R.A. Changing weather extremes call for early warning of potential for catastrophic fire. *Earth's Future* **2017**, *5*, 1196–1202. <https://doi.org/10.1002/2017EF000657>.
12. Dahanayake, K.C.; Chow, C.L. Moisture content, ignitability, and fire risk of vegetation in vertical greenery systems. *Fire Ecol.* **2018**, *14*, 125–142. <https://doi.org/10.4996/fireecology.140112514>.
13. Capps, S.B.; Zhuang, W.; Liu, R.; Rolinski, T.; Qu, X. Modelling chamise fuel moisture content across California: A machine learning approach. *Int. J. Wildland Fire* **2021**, *31*, 136–148. <https://doi.org/10.1071/WF21061>.
14. Castro, F.; Tudela, A.; Sebastià, M.T. Modeling moisture content in shrubs to predict fire risk in Catalonia (Spain). *Agric. For. Meteorol.* **2003**, *116*, 49–59.
15. Dennison, P.E.; Moritz, M.A. Critical live fuel moisture in chaparral ecosystems: A threshold for fire activity and its relationship to antecedent precipitation. *Int. J. Wildland Fire* **2009**, *18*, 1021–1027.
16. Dimitrakopoulos, P.A.; Bemmerzouk, A.M. Predicting live herbaceous moisture content from a seasonal drought index. *Int. J. Biometeorol.* **2003**, *47*, 73–79.
17. Holden, A.Z.; Jolly, W.M. Modeling topographic influences on fuel moisture and fire danger in complex terrain to improve wildland fire management decision support. *For. Ecol. Manag.* **2011**, *262*, 2133–2141.

18. Nolan, R.H.; Boer, M.M.; de Dios, V.R.; Caccamo, G.; Bradstock, R.A. Large-scale, dynamic transformations in fuel moisture drive wildfire activity across southeastern Australia. *Geophys. Res. Lett.* **2016**, *43*, 4229–4238.
19. Pellizzaro, G.; Duce, P.; Ventura, A.; Zara, P. Seasonal variations of live moisture content and ignitability in shrubs of the Mediterranean Basin. *Int. J. Wildland Fire* **2007**, *16*, 633–641.
20. Pivovarov, A.L.; Emery, N.; Sharifi, M.R.; Witter, M.; Keeley, J.E.; Rundel, P.W. The effect of ecophysiological traits on live fuel moisture content. *Fire* **2019**, *2*, 28.
21. Ruffault, J.; Martin-StPaul, N.; Pimont, F.; Dupuy, J.-L. How well do meteorological drought indices predict live fuel moisture content (LFMC)? An assessment for wildfire research and operations in Mediterranean ecosystems. *Agric. For. Meteorol.* **2018**, *262*, 391–401.
22. Chou, D.M.; Suarez, M.J. A Solar Radiation Parameterization (CLIRAD-SW) Developed at Goddard Climate and Radiation Branch for Atmospheric Studies. NASA Technical Memorandum U.S. Patent NASA/TM-1999-104606 15, 1 June 1999.
23. Chuvieco, E.; Aguado, I.; Dimitrakopoulos, A.P. Conversion of fuel moisture content values to ignition potential for integrated fire danger assessment. *Can. J. For. Res.* **2004**, *34*, 2284–2293.
24. Danson, F.; Bowyer, P. Estimating live fuel moisture content from remotely sensed reflectance. *Remote Sensing of Environment* **2004**, *92*, 309–321.
25. Jurdao, S.; Chuvieco, E.; Arevalillo, J.M. Modelling fire ignition probability from satellite estimates of live fuel moisture content. *Fire Ecol.* **2012**, *8*, 77–97.
26. Myoung, B.; Kim, S.H.; Nghiem, S.V.; Jia, S.; Whitney, K.; Kafatos, M.C. Estimating live fuel moisture from MODIS satellite data for wildfire danger assessment in Southern California USA. *Remote Sens.* **2018**, *10*, 87.
27. Peterson, S.H.; Roberts, D.A.; Dennison, P.E. Mapping live fuel moisture with MODIS data: A multiple regression approach. *Remote Sens. Environ.* **2008**, *112*, 4272–4284.
28. Qi, Y.; Dennison, P.E.; Spencer, J.; Riaño, D. Monitoring live fuel moisture using soil moisture and remote sensing proxies. *Fire Ecol.* **2012**, *8*, 71–87.
29. Rao, K.; Williams, A.P.; Flefil, J.F.; Konings, A.G. SAR-enhanced mapping of live fuel moisture content. *Remote Sens. Environ.* **2020**, *245*, 111797.
30. Yebra, M.; Dennison, P.E.; Chuvieco, E.; Riaño, D.; Zylstra, P.; Hunt, E.R., Jr.; Danson, F.M.; Qi, Y.; Jurdao, S. A global review of remote sensing of live fuel moisture content for fire danger assessment: Moving towards operational products. *Remote Sens. Environ.* **2013**, *136*, 455–468.
31. Yebra, M.; Quan, X.; Riaño, D.; Larraondo, P.R.; van Dijk, A.I.; Cary, G.J. A fuel moisture content and flammability monitoring methodology for continental Australia based on optical remote sensing. *Remote Sens. Environ.* **2018**, *212*, 260–272.
32. Jain, P.; Coogan, S.C.P.; Subramanian, S.G.; Crowley, M.; Taylor, S.W.; Flannigan, M.D. A review of machine learning applications in wildfire science and management. *Environ. Rev.* **2020**, *28*, 478–505.
33. McCandless, T.C.; Kosovic, B.; Petzke, W. Enhancing wildfire spread modelling by building a gridded fuel moisture content product with machine learning. *Mach. Learn. Sci. Technol.* **2020**, *1*, 035010.
34. Michael, Y.; Helman, D.; Glickman, O.; Gabay, D.; Brenner, S.; Lensky, I.M. Forecasting fire risk with machine learning and dynamic information derived from satellite vegetation index time-series. *Sci. Total Environ.* **2021**, *764*, 142844.
35. Corona, P.; Ascoli, D.; Barbati, A.; Bovio, G.; Colangelo, G.; Elia, M.; Garfi, V.; Iovino, F.; Laforteza, R.; Leone, V.; et al. Integrated forest management to prevent wildfires under Mediterranean environments. *Annals of Silvicultural Research*. 2015, 39 (1), 1–22. <https://doi.org/10.12899/asr-946>.
36. Schultz, C.A.; Thompson, M.P.; McCaffrey, S.M. Forest Service fire management and the elusiveness of change. *Fire Ecol.* **2019**, *15*, 13. <https://doi.org/10.1186/s42408-019-0028-x>.

37. Dimitrakopoulos, A.P.; Bemmerzouk, A.M.; Mitsopoulos, I.D. Evaluation of the Canadian fire weather index system in an eastern Mediterranean environment. *Meteorol. Appl.* **2011**, *18*, 83–93.
38. Wang, X.; Wotton, B.M.; Cantin, A.S.; Parisien, M.-A.; Anderson, K.; Moore, B.; Flannigan, M.D. cffdrs: An R package for the Canadian forest fire danger rating system. *Ecol. Processes* **2017**, *6*, 5. <https://doi.org/10.1186/s13717-017-0070-z>.
39. Ye, T.; Wang, Y.; Guo, Z.; Li, Y. Factor contribution to fire occurrence, size, and burn probability in a subtropical coniferous forest in East China. *PLoS ONE* **2017**, *12*, e0172110.
40. Chas-Amil, M.L.; Touza, J.M.; Prestemon, J.P.; McClean, C.J. Natural and social factors influencing forest fire occurrence at a local spatial scale. In *Modelling Fire Behavior and Risk*; Donatella, S., Valentina, B., Michele, S., Costatino, S., Eds.; Global Fire Monitoring Center: Freiburg, Germany, 2012; pp. 181–186.
41. Romps, D.M.; Seeley, J.T.; Vollaro, D.; Molinari, J. Projected increase in lightning strikes in the United States due to global warming. *Science* **2014**, *346*, 851–854. <https://doi.org/10.1126/science.1259100>.
42. Krause, A.; Kloster, S.; Wilkenskjeld, S.; Paeth, H. The sensitivity of global wildfires to simulated past, present, and future lightning frequency. *J. Geophys. Res. Biogeosciences* **2014**, *119*, 312–322.
43. Quílez, R.; Merida, E. Manual de Seguridad en Operaciones de Extinción de Incendios Forestales; *Pau Costa Foundation*, Ed.; Barcelona, Spain; 333p, ISBN 978-84-617-1323-3.
44. Karavani, A.; Boer, M.M.; Baudena, M.; Colinas, C.; Díaz-Sierra, R.; Pemán, J.; de Luis, M.; Enriquez-de-Salamanca, A.; Resco de Dios, V. Fire-induced deforestation in drought-prone Mediterranean forests: Drivers and unknowns from leaves to communities. *Ecol. Monogr.* **2018**, *88*, 141–169.
45. Nolan, R.H.; Blackman, C.J.; de Dios, V.R.; Choat, B.; Medlyn, B.E.; Li, X.; Bradstock, R.A.; Boer, M.M. Linking forest flammability and plant vulnerability to drought. *Forests* **2020**, *11*, 779. <https://doi.org/10.3390/f11070779>.
46. Pellizzaro, G.; Cesaraccio, C.; Duce, P.; Ventura, A.; Zara, P. Relationships between seasonal patterns of live fuel moisture and meteorological drought indices for Mediterranean shrubland species. *Int. J. Wildland Fire* **2007**, *16*, 232–241.
47. Turco, M.; von Hardenberg, J.; AghaKouchak, A.; Llasat, M.C.; Provenzale, A.; Trigo, R.M. On the key role of droughts in the dynamics of summer fires in Mediterranean Europe. *Sci. Rep.* **2017**, *7*, 81. <https://doi.org/10.1038/s41598-017-00116-9>.
48. Boer, M.M.; De Dios, V.R.; Bradstock, R.A. Unprecedented burn area of Australian mega forest fires. *Nat. Clim. Chang.* **2020**, *10*, 171–172.
49. Reddy, P.J.; Sharples, J.J.; Lewis, S.C.; Perkins-Kirkpatrick, S.E. Modulating influence of drought on the synergy between heatwaves and dead fine fuel moisture content of bushfire fuels in the Southeast Australian region. *Weather Clim. Extrem.* **2021**, *31*, 100300. <https://doi.org/10.1016/j.wace.2020.100300>.
50. Littell, J.S.; Peterson, D.L.; Riley, K.L.; Liu, Y.; Luce, C. A review of the relationships between drought and forest fire in the United States. *Glob. Chang. Biol.* **2016**, *22*, 2353–2369. <https://doi.org/10.1111/gcb.13275>.
51. Monroe, D.C.; Blumenfeld, R.S.; Keator, D.B.; Solodkin, A.; Small, S.L. Post-fire regeneration variability of *Pinus halepensis* in the eastern Iberian Peninsula. *For. Ecol. Manag.* **2004**, *203*, 251–259. <https://doi.org/10.1016/J.FORECO.2004.07.061>.
52. Varol, T.; Ertugrul, M.; Özel, H.B. Drought-Forest Fire Relationship. *J. Mediterr. Identities—Environ. Soc. Cult.* **2017**, 283–303. <https://doi.org/10.5772/intechopen.68487>.
53. Quílez, R.; Valbuena, L.; Vendrell, J.; Uytewaal, K.; Ramirez, J. Establishing Propagation Nodes as a Basis for Preventing Large Wildfires: The Proposed Methodology. *Front. For. Glob. Chang.* **2020**, *3*, 137. <https://doi.org/10.3389/ffgc.2020.548799>.
54. Thornthwaite, C.W. An approach toward a rational classification of climate. *Geogr. Rev.* **1948**, *38*, 55–94.

55. Nesterov, V. *Forest Fires and Methods of Fire Risk Determination*; Goslesbumizdat: Moscow, Russian, 1949.
56. Käse, H. *Ein Vorschlag für eine Methode zur Bestimmung und Vorhersage der Waldbrandgefährdung mit Hilfe komplexer Kennziffern*; Akademie-Verlag: Berlin, Germany, 1969.
57. Reinhard, M.; Rebetetz, M.; Schlaepfer, R. Recent climate change: Rethinking drought in the context of forest fire research in Ticino, South of Switzerland. *Theor. Appl. Climatol.* **2005**, *82*, 17–25.
58. Sancho, J.L.; Moraga, R.Q. Análisis de la humedad del combustible vivo en la Comunitat Valenciana. In *Actas del 7º Congreso Forestal Español*; *Sociedad Española de Ciencias Forestales*: Plasencia, Spain, 2017.
59. Sardans, J.; Peñuelas, J. Plant-soil interactions in Mediterranean forest and shrublands: Impacts of climatic change. *Plant Soil* **2013**, *365*, 1–33. <https://doi.org/10.1007/s11104-013-1591-6>.
60. Wilson, N.; Bradstock, R.; Bedward, M. Detecting the effects of logging and wildfire on forest fuel structure using terrestrial laser scanning (TLS). *For. Ecol. Manag.* **2021**, *488*, 119037. <https://doi.org/10.1016/j.foreco.2021.119037>.
61. Zapata, R.; Oliver-Villanueva, J.-V.; Lemus-Zúñiga, L.-G.; Fuente, D.; Pla, M.A.M.; Luzuriaga, J.E.; Esteve, J.C.M. Seasonal variations of electrical signals of *Pinus halepensis* Mill. in Mediterranean forests in dependence on climatic conditions. *Plant Signal. Behav.* **2021**, *16*, 1948744. <https://doi.org/10.1080/15592324.2021.1948744>.
62. Gale, M.G.; Cary, G.J.; Van Dijk, A.I.; Yebra, M. Forest fire fuel through the lens of remote sensing: Review of approaches, challenges and future directions in the remote sensing of biotic determinants of fire behaviour. *Remote Sens. Environ.* **2021**, *255*, 112282. <https://doi.org/10.1016/j.rse.2020.112282>.
63. Rothermel, R.C. *How to Predict the Spread and Intensity of Forest and Range Fires*; US Department of Agriculture, Forest Service, Intermountain Forest and Range Experiment Station: Minneapolis, MN, USA, 1983; Volume 143.
64. Castro, F.X.; Tudela, A.; Gabriel, E.; Montserrat, D.; Canyameres, E.; Segarra, M. Evolution of live fuel moisture in Mediterranean forest. *For. Ecol. Manag.* **2006**, *234*, S34.
65. Viegas, D.X.; Soares, J.; Almeida, M. Combustibility of a mixture of live and dead fuel components. *Int. J. Wildland Fire* **2013**, *22*, 992–1002.
66. Schroeder, M.J.; Buck, C.C. *Fire Weather, Agricultural Handbook 360*; USDA Forest Service: Washington, DC, USA, 1970.
67. Countryman, C.M. *Measuring Moisture Content in Living Chaparral: A Field User's Manual (Vol. 36)*; US Department of Agriculture, Forest Service, Pacific Southwest Forest and Range Experiment Station: Prather, CA, USA, 1979.
68. de Luis, M.; Čufar, K.; Di Filippo, A.; Novak, K.; Papadopoulos, A.; Piovesan, G.; Rathgeber, C.B.; Raventos, J.; Saz, M.A.; Smith, K.T. Plasticity in dendroclimatic response across the distribution range of Aleppo pine (*Pinus halepensis*). *PLoS ONE* **2013**, *8*, e83550. <https://doi.org/10.1371/journal.pone.0083550>.
69. AAVV. Distribution Map of Aleppo Pine. EUFORGEN 2009, 2008. Available online: [www.euforgen.org](http://www.euforgen.org) (accessed 16 July 2020 July).
70. Fady, B.; Semerci, H.; Vendramin, G.G. *EUFORGEN Technical Guidelines for genetic conservation and use for Aleppo pine (Pinus halepensis) and Brutia pine (Pinus brutia)*; Bioversity International: Rome, Italy, 2003; p. 6, ISBN 92-9043-571-2.
71. Mauri, A.; Di Leo, M.; De Rigo, D.; Caudullo, G. *Pinus halepensis* and *Pinus brutia* in Europe: Distribution, habitat, usage and threats. In *European Atlas of Forest Tree Species*; Publications Office of the EU: Luxembourg, 2016;
72. IFN3. *Tercer Inventario Forestal Nacional (3rd National Forest Inventory of Spain)*; Ministerio para la Transformación Ecológica y el Reto Demográfico: Madrid, Spain, 2007.
73. Smith, D.; Allen, S. Measurement of sap flow in plant stems. *J. Exp. Bot.* **1996**, *47*, 1833–1844.

74. Park, H.J.; Park, J.H.; Park, K.S.; Ahn, T.I.; Son, J.E. Nondestructive measurement of paprika (*Capsicum annuum* L.) internal electrical conductivity and its relation to environmental factors. *Horticultural Science and Technology*. 31 October 2018. 691-701 <https://doi.org/10.12972/kjhst.20180069>.
75. Scarff, F.R.; Lenz, T.; Richards, A.E.; Zanne, A.E.; Wright, I.J.; Westoby, M. Effects of plant hydraulic traits on the flammability of live fine canopy fuels. *Funct. Ecol.* **2021**, *35*, 835–846. <https://doi.org/10.1111/1365-2435.13771>.
76. Hao, Z.; Li, W.; Hao, X. Variations of electric potential in the xylem of tree trunks associated with water content rhythms. *J. Exp. Bot.* **2021**, *72*, 1321–1335.
77. Love, C.; Zhang, S.; Mershin, A. Source of sustained voltage difference between the xylem of a potted *Ficus benjamina* tree and its soil. *PLoS ONE* **2008**, *3*, e2963.
78. Burdon-Sanderson, J.S. I. Note on the electrical phenomena which accompany irritation of the leaf of *Dionaea muscipula*. *Proc. R. Soc. Lond.* **1873**, *21*, 495–496.
79. Darwin, C. *Insectivorous Plants*; D Appleton & Company: New York, NY, USA, 1875. <https://doi.org/10.5962/bhl.title.99933>.
80. Wright, J.P.; Fisher, D.B. Measurement of the sieve tube membrane potential. *Plant Physiol.* **1981**, *67*, 845–848. <https://doi.org/10.1104/PP.67.4.8456>.
81. Gibert, D.; Le Mouël, J.-L.; Lambs, L.; Nicollin, F.; Perrier, F. Sap flow and daily electric potential variations in a tree trunk. *Plant Sci.* **2006**, *171*, 572–584. <https://doi.org/10.1016/j.plantsci.2006.06.012>.
82. Oyarce, P.; Gurovich, L. Electrical signals in avocado trees: Responses to light and water availability conditions. *Plant Signal. Behav.* **2010**, *5*, 34–41. <https://doi.org/10.4161/psb.5.1.10157>.
83. Gil, P.M.; Gurovich, L.; Schaffer, B. The electrical response of fruit trees to soil water availability and diurnal light-dark cycles. *Plant Signal. Behav.* **2008**, *3*, 1026–1029. <https://doi.org/10.4161/psb.6786>.
84. Gil, P.M.; Gurovich, L.; Schaffer, B.; García, N.; Iturriaga, R. Electrical signaling, stomatal conductance, ABA and ethylene content in avocado trees in response to root hypoxia. *Plant Signal. Behav.* **2009**, *4*, 100–108. <https://doi.org/10.4161/psb.4.2.7872>.
85. Rios-Rojas, L.; Morales-Moraga, D.; Alcalde, J.; Gurovich, L. Use of plant woody species electrical potential for irrigation scheduling. *Plant Signal. Behav.* **2015**, *10*, e976487. <https://doi.org/10.4161/15592324.2014.976487>.
86. Fromm, J.; Lautner, S. Electrical signals and their physiological significance in plants. *Plant Cell Environ.* **2007**, *30*, 249–257.
87. The Editors of Encyclopaedia Britannica. *Electric Power*; Encyclopædia Britannica; Encyclopædia Britannica, Inc.: Chicago, IL, USA. Available online: <https://www.britannica.com/technology/electric-power> (accessed 31 August 2020).
88. Korol, L.; Madmony, A.; Riov, Y.; Schiller, G. *Pinus halepensis* × *Pinus brutia* subsp. *brutia* hybrids? Identification using morphological and biochemical traits. *Silvae Genet.* **1995**, *44*, 186–190.
89. Allard, G.; Berrahmouni, N.; Besacier, C.; Boglio, D.; Briens, M.; Brizay, A.; Camia, A.; Colletti, L.; Conigliaro, M.; D'Annunzio, R. et al. State of Mediterranean Forests 2013; *FAO*, **2013** Rome, Italy.
90. Buhk, C.; Meyn, A.; Jentsch, A. The challenge of plant regeneration after fire in the Mediterranean Basin: Scientific gaps in our knowledge on plant strategies and evolution of traits. *Plant Ecol.* **2007**, *192*, 1–19.
91. Heras, J.D.; Moya, D.; Vega, J.A.; Daskalakou, E.; Vallejo, V.R.; Grigoriadis, N.; Tsitsoni, T.; Baeza, J.; Valdecantos, A.; Fernández, C.; et al. Post-fire management of serotinous pine forests. In *Post-Fire Management and Restoration of Southern European Forests*; Springer: Dordrecht, The Netherlands, 2012; pp. 121–150.
92. Vallejo, M.; Arianoutsou, F.M. *Post-Fire Management and Restoration of Southern European Forests*; Moreira, F., Arianoutsou, M., Corona, P., De las Heras, J., Eds.; Managing Forest Ecosystems; Springer: Dordrecht, The Netherlands, 2012; Volume 24, pp. 93–119.
93. Farjon, A. *A Handbook of the World's Conifers*; Brill: Leiden, The Netherlands, 2010.

94. Rathgeber, C.; Nicault, A.; Guiot, J.; Keller, T.; Guibal, F.; Roche, P. Simulated responses of *Pinus halepensis* forest productivity to climatic change and CO<sub>2</sub> increase using a statistical model. *Glob. Planet. Chang.* **2000**, *26*, 405–421. [https://doi.org/10.1016/S0921-8181\(00\)00053-9](https://doi.org/10.1016/S0921-8181(00)00053-9).
95. Thuiller, W. BIOMOD—optimizing predictions of species distributions and projecting potential future shifts under global change. *Glob. Chang. Biol.* **2003**, *9*, 1353–1362.
96. Urli, M.; Delzon, S.; Eyermann, A.; Couallier, V.; García-Valdés, R.; Zavala, M.A.; Porté, A.J. Inferring shifts in tree species distribution using asymmetric distribution curves: A case study in the Iberian mountains. *J. Veg. Sci.* **2014**, *25*, 147–159.
97. Osem, Y.; Lavi, A.; Rosenfeld, A. Colonization of *Pinus halepensis* in Mediterranean habitats: Consequences of afforestation, grazing and fire. *Biol. Invasions* **2011**, *13*, 485–498. <https://doi.org/10.1007/s10530-010-9843-3>.
98. Zapata, R.; Oliver-Villanueva, J.-V.; Lemus-Zúñiga, L.-G.; Luzuriaga, J.E.; Pla, M.A.M.; Urchueguía, J.F. Evaluation of electrical signals in pine trees in a mediterranean forest ecosystem. *Plant Signal. Behav.* **2020**, *15*, 1795580. <https://doi.org/10.1080/15592324.2020.1795580>.
99. Hapla, F.; Saborowski, J. Planning of sample size for wood anatomical investigations. *Holz Als Roh-Und Werkst.* **1987**, *45*, 141–144.
100. Oliver-Villanueva, J.V.; Becker, G. Verwendungsrelevante Holzeigenschaften der Esche (*Fraxinus excelsior* L.) und ihre Variabilität im Hinblick auf Alter und Standraum. *Forst Und Holz* **1993**, *48*, 387–391.
101. Hapla, F.; Oliver-Villanueva, J.V.; González-Molina, J.M. Effect of silvicultural management on wood quality and timber utilisation of *Cedrus atlantica* in the European Mediterranean area. *Holz Als Roh-Und Werkst.* **2000**, *58*, 1–8. <https://doi.org/10.1007/s001070050377>.
102. Agencia Estatal de Meteorología (Niveles de Riesgo de Incendio del Sistema FWI AEMET) España. <https://prevencionincendiosgva.es/Meteorologia/InformesPrevisiones> (accessed on 15 December 2021).
103. Volkov, A.G.; Ranatunga, D.R.A. Plants as environmental biosensors. *Plant Signal Behav.* **2006**, *1*, 105–115.
104. Cardoso, S.S.L.B.; Carrondo, J.M.; Marques, P.N.; Narciso, M.J.; Rocha, I.N.; Soares, R.A. Monitorization of the electrical signal generated by a tree. *Proceedings of 4th Luso-spanish assembly on geodesy and geophysics* Figueira da Foz, Portugal. 2004.
105. Directive 1999/5/Ec of the European Parliament and of the council of 9 march 1999.
106. Madrigal, J.; Hernando, C.; Guijarro, M. A new bench-scale methodology for evaluating the flammability of live forest fuels. *J. Fire Sci.* **2013**, *31*, 131–142. <https://doi.org/10.1177/0734904112458244>.
107. SALTUS Spot fires; mechanisms, análisis and modeling. Technical Annex. Commission of the European Communities *Instituto Nacional de Investigación y Tecnología Agraria y Alimentaria Laboratorio del Fuego*; Ministerio de Agricultura, Gobierno de España: Madrid, Spain. 20p 1997
108. Asociación Valenciana de Meteorología ‘Josep Peinado’ (AVAMET). Available online: <https://www.avamet.org/mx-mxo.php?id=c11m902e01>. (accessed on 20 December 2021).
109. Costa-Saura, J.M.; Balaguer-Beser, Á.; Ruiz, L.A.; Pardo-Pascual, J.E.; Soriano-Sancho, J.L. Empirical Models for Spatio-Temporal Live Fuel Moisture Content Estimation in Mixed Mediterranean Vegetation Areas Using Sentinel-2 Indices and Meteorological Data. *Remote Sens.* **2021**, *13*, 3726.
110. Mitsopoulos, I.D.; Dimitrakopoulos, A.P. Allometric equations for crown fuel biomass of Aleppo pine (*Pinus halepensis* Mill.) in Greece. *Int. J. Wildland Fire* **2007**, *16*, 642–647. <https://doi.org/10.1071/WF06038>.
111. Matamoros, M.R.; Merino, E.G.; Ibáñez, N.I.; Bernal, E.M. Sensibilidad y grado de adaptación de “*Pinus halepensis*” mill. a la variabilidad climática en la provincia de Zaragoza. *Cuad. De La Soc. Española De Cienc. For.* **2008**, *26*, 137–142.

112. Puertolas, J.; Sierra, R.; Pardos, J.A. Comportamiento fisiológico de una plantación de *Pinus halepensis* y *Pinus pinea* en un antiguo terreno agrícola. Proceedings of *IV Congreso Forestal Español. Sociedad Española de Ciencias Forestales*, Zaragoza, Spain 2005.
113. Fotelli, M.N.; Korakaki, E.; Paparrizos, S.A.; Radoglou, K.; Awada, T.; Matzarakis, A. Environmental controls on the seasonal variation in gas exchange and water balance in a near-coastal Mediterranean *Pinus halepensis* forest. *Forests* **2019**, *10*, 313.
114. Sánchez-Costa, E.; Poyatos, R.; Sabaté, S. Contrasting growth and water use strategies in four co-occurring Mediterranean tree species revealed by concurrent measurements of sap flow and stem diameter variations. *Agric. For. Meteorol.* **2015**, *207*, 24–37.
115. Keane, R.E. *Wildland Fuel Fundamentals and Applications*; Springer International Publishing: New York, NY, USA, 2015; ISBN 9783319090153.
116. de Dios Rinaudo, R. *Plant-Fire Interactions*, 1st ed.; Applying Ecophysiology to Wildfire Management (Managing Forest Ecosystems, 36); Springer: Cham, Switzerland, 2020; ISBN 9783030411916.
117. Alexander, M.E.; Cruz, M.G. Corrigendum to: Assessing the effect of foliar moisture on the spread rate of crown fires. *Int. J. Wildland Fire* **2013**, *22*, 869–870.
118. Jenkins, M.J.; Page, W.G.; Hebertson, E.G.; Alexander, M.E. Fuels and fire behavior dynamics in bark beetle-attacked forests in Western North America and implications for fire management. *For. Ecol. Manag.* **2012**, *275*, 23–34.
119. Talucci, A.C.; Krawchuk, M.A. Dead forests burning: The influence of beetle outbreaks on fire severity and legacy structure in sub-boreal forests. *Ecosphere* **2019**, *10*, e02744. <https://doi.org/10.1002/ecs2.2744>.
120. Reiner, A.L. *Fire Behavior in Beetle-Killed Stands: A Brief Review of Literature Focusing on Early Stages after Beetle Attack*; US Forest Service Pacific Southwest Region: Vallejo, CA, USA, 2017; Volume 5, pp. 1–5.
121. Martín, M.S.; Bonet, J.A.; De Aragón, J.M.; Voltas, J.; Coll, L.; De Dios, V.R. Crown bulk density and fuel moisture dynamics in *Pinus pinaster* stands are neither modified by thinning nor captured by the Forest Fire Weather Index. *Ann. For. Sci.* **2017**, *74*, 51. <https://doi.org/10.1007/s13595-017-0650-1>.
122. Nolan, R.H.; Hedro, J.; Arteaga, C.; Sugai, T.; de Dios, V.R. Physiological drought responses improve predictions of live fuel moisture dynamics in a Mediterranean forest. *Agric. For. Meteorol.* **2018**, *263*, 417–427. <https://doi.org/10.1016/j.agrformet.2018.09.011>.
123. Klein, T.; Cohen, S.; Yakir, D. Hydraulic adjustments underlying drought resistance of *Pinus halepensis*. *Tree Physiol.* **2011**, *31*, 637–648. <https://doi.org/10.1093/treephys/tp1047>.
124. Klein, T.; Cohen, S.; Paudel, I.; Preisler, Y.; Rotenberg, E.; Yakir, D. Diurnal dynamics of water transport, storage and hydraulic conductivity in pine trees under seasonal drought. *Forest-Biogeosciences For.* **2016**, *9*, 710. <https://doi.org/10.3832/for2046-009>.
125. Oliveras, I.; Martínez-Vilalta, J.; Jimenez-Ortiz, T.; Lledó, M.J.; Escarré, A.; Pinol, J. Hydraulic properties of *Pinus halepensis*, *Pinus pinea* and *Tetraclinis articulata* in a dune ecosystem of Eastern Spain. *Plant Ecol.* **2003**, *169*, 131–141. <https://doi.org/10.1023/A:1026223516580>.
126. Pacheco, A.; Camarero, J.J.; Ribas, M.; Gazol, A.; Gutierrez, E.; Carrer, M. Disentangling the climate-driven bimodal growth pattern in coastal and continental Mediterranean pine stands. *Sci. Total Environ.* **2018**, *615*, 1518–1526. <https://doi.org/10.1016/j.scitotenv.2017.09.133>.
127. Prislán, P.; Gričar, J.; de Luis, M.; Novak, K.; del Castillo, E.M.; Schmitt, U.; Koch, G.; Štrus, J.; Mrak, P.; Žnidarič, M.T.; et al. Annual cambial rhythm in *Pinus halepensis* and *Pinus sylvestris* as indicator for climate adaptation. *Front. Plant Sci.* **2016**, *7*, 1923. <https://doi.org/10.3389/fpls.2016.01923>.
128. Schiller, G.; Cohen, Y. Water regime of a pine forest under a Mediterranean climate. *Agric. For. Meteorol.* **1995**, *74*, 181–193. [https://doi.org/10.1016/0168-1923\(94\)02195-P](https://doi.org/10.1016/0168-1923(94)02195-P).
129. Rossa, C.G. A generic fuel moisture content attenuation factor for fire spread rate empirical models. *For. Syst.* **2018**, *27*, e009. <https://doi.org/10.5424/fs/2018272-13175>.

130. Moraga, R.Q.; Cisneros, J.R.; Relea, M.L. Prevención de Megaincendios Forestales Mediante el Diseño de Planes de Operaciones de Extinción Basados en Nodos de Propagación. Ph.D. Dissertation, Universidad de León, León, Spain, 2016.
131. Qi, Y.; Dennison, P.E.; Jolly, W.M.; Kropp, R.C.; Brewer, S.C. Spectroscopic analysis of seasonal changes in live fuel moisture content and leaf dry mass. *Remote Sens. Environ.* **2014**, *150*, 198–206.
132. Balaguer-Romano, R.; Díaz-Sierra, R.; Madrigal, J.; Voltas, J.; Resco de Dios, V. Needle senescence affects fire behavior in Aleppo pine (*Pinus halepensis* Mill.) stands: A simulation study. *Forests* **2020**, *11*, 1054.
133. Mukaka, M.M. A guide to appropriate use of correlation coefficient in medical research. *Malawi Med. J.* **2012**, *24*, 69–71.

# Biomass assessment in homogeneous pine forest stands using Landsat images and biomass equations in the province of Palencia (Spain)

Vinué-Visús, D.<sup>1\*</sup>, Coll-Aliaga, E.<sup>1</sup>, Oliver-Villanueva, J.V.<sup>1</sup>, Lerma-Arce, V.<sup>1</sup>, Lorenzo-Sáez, E.<sup>1</sup> Fuente, D.<sup>2</sup>

<sup>1</sup>ITACA Research Institute, Universitat Politècnica de València, 46022 València, Spain;

\*Correspondence to [davivil@upv.es](mailto:davivil@upv.es)

<sup>2</sup>CzechGlobe - Global Change Research Centre, Academy of Science of the Czech Republic, 66424 Drásov, Czech Republic; [fuentes.d@czechglobe.cz](mailto:fuentes.d@czechglobe.cz)

**Abstract.** In this contribution, we assessed the biomass of two types of pine forest in North of Spain, including *Pinus sylvestris* and *Pinus nigra* by means of forest inventories and Landsat imagery where variables were linked using non-linear, non-parametric Gaussian process regression methods. Although biomass means for every type of forest is similar (59.80 and 56.38 Mg·ha<sup>-1</sup> respectively), forest cannot be considered as an only class and must be segmented by species and by age. In that form, avoiding confusions, with a reduced inventory of only 15 plots for *Pinus nigra* covering up to 1,900 ha, the total biomass could be assessed with a root-mean-square error of 13.97 Mg·ha<sup>-1</sup> and a bias of 0.36 Mg·ha<sup>-1</sup> for a range of values between 40.22 and 83.91 Mg·ha<sup>-1</sup> getting a relative error between 15.6% and 34.7% for the gathered biomass. This is a rather good estimation considering the little effort and time invested; thus, the suggested methodology is very suitable for forest monitoring and management.

## 1 Introduction

Biomass inventories are the key in the design and evaluation of policies related to carbon stocks and carbon sequestration including forest management plans to maximize the protection and the positive dynamics of both carbon stocks and carbon sequestration [1]. Consequently, there is a need for robust carbon accounting [2].

For that purpose, and in terms of field sampling, the assessment of biomass and thus of carbon stocks can be achieved on a large scale through traditional forest inventories, such as the National Forest Inventory (NFI) in Spain and by using biomass equations [3]. However, the inventoried areas and the distance between plots in the NFI (typically 1 km) are quite large as both are designed for a global assessment. In addition, the time between consecutive fieldwork, i.e., the actual time lag is already more than 15 years for most of the Spanish territory meaning a lack of accurate and updated information in specific areas of interest.

For large and widespread areas, some methodologies have been proposed for the assessment of biomass. However, these make use of general equations without distinguishing between species. In this sense, large areas have been studied with a general classification regardless of the species and with a general classification independent of its specific composition [4].

From a remote sensing point of view, ensuring ground-truth is essential: although the relationship between reflectance in the optical range and biomass has been studied, it is not possible to correctly assess biomass from remotely sensed imagery without linking this information to field data [5] [6]. It means that, for any study, field data must be taken for calibration and validation purposes.

Regardless of the origin of remote sensing imagery, both from passive and active sensors, several methodologies for data processing have been tested, including the active sensors, several methodologies have been tested for data processing, including multiple regression analysis, k-NN methods and neural networks, among others [7]. Other empirical methods are based on non-parametric regression functions without any explicit assumptions about the dependencies of the variables or the distribution of the data. These methods avoid carrying out prior relations of spectral bands, transformations, or fitting functions [8].

Empirical methods are limited within the range of values of the training dataset, and it is therefore difficult to extrapolate the results to other conditions or biomes [9]. However, some of the non-parametric regression methods have demonstrated their adaptability to remote sensing studies. Among them, non-linear machine-learning regression algorithms, such as artificial neural networks (ANNs), Support Vector Machines (SVMs) or Gaussian Processes Regression (GPR) have been effectively applied for the assessment of biophysical variables from Earth observation data [10] [11].

GPRs are a collection of finite random variables with a multivariate normal [12]. These processes are related to a collection of indexed random variables that can be defined through a shared probability density, typically a Gaussian distribution. The application of GPRs for biomass assessment have not been tested in depth so far. In addition, those previous works described above use a wide range of values and field data and consequently cannot be taken as a reference to estimate the number of samples needed for a good performance.

## 2 Objective

The general objective of this work is to estimate the biomass of two types of pine forest in the province of Palencia, Spain.

The specific objectives are (i) to define a general algorithm using an empirical method based on non-parametric method based on automatic learning techniques of Gaussian processes to be applied to Landsat imagery; (ii) to evaluate the capacity of the specific

algorithms to estimate the total biomass of two types of characteristic pine vegetation in the province of Palencia.

### 3 Methodology

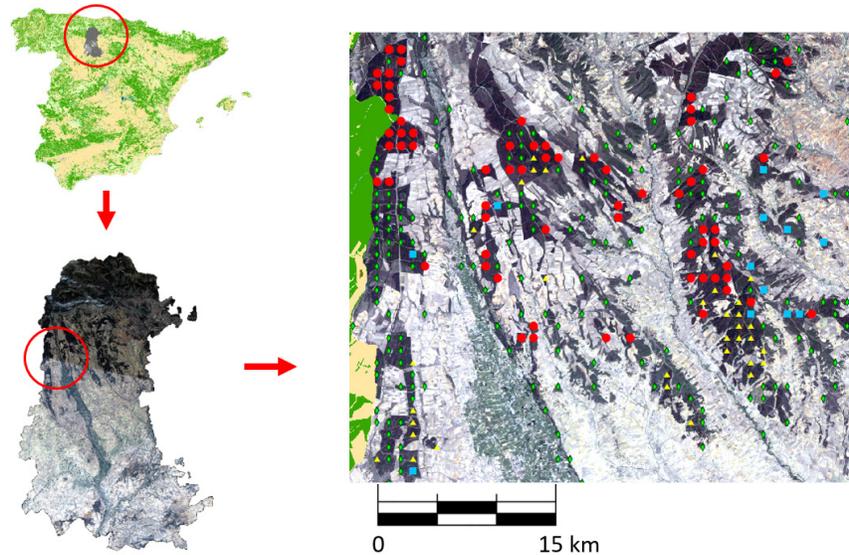
#### 3.1 Study Area

The study focuses on the province of Palencia, in the natural region of “Páramos y Valles palentinos”, which is the link between the high peaks of the “Montaña palentina” and the extensive plains of “Tierra de Campos”, with an average altitude between 800 and 1,000 metres above sea level.

The landscape is characterized by inter-mountainous, undulating terrain, with series of high moorlands and wide valleys. In terms of climate, the character of the area as a transition zone between the Mountains and Tierra de Campos reflects a slight Atlantic influence and the characteristics of the Mediterranean area (Fig. 1).

The natural vegetation of this region is made up of scrub oak (*Quercus pyrenaica* Willd.), together with some holm oak (*Quercus ilex* L.) and gall oak (*Quercus faginea* Lam.) in the most arid areas. The typical ecosystem of this area has been transformed by various alterations such as the extraction of firewood, fires, grazing, clearing of the forest for agricultural and livestock use and, above all, by reforestation, which began mainly in the 1960s.

These species are accompanied by poplars and elms, shrubs such as heather, broom, wild roses, hawthorns, blackthorns, and gorse. The Scots pine (*Pinus sylvestris* L.), and black pines (*Pinus pinaster* Ait. and *Pinus nigra* Arn.) are reforestation that were established for soil regeneration and protection against erosion of slopes and gullies in the 1960s. In the last 15-20 years before the last inventory of the province, Third National Forest Inventory in 2011 (NFI3), they have been cleared and thinned. Finally, the soils are mainly siliceous.



**Fig. 1.** Study area and location of field plots as in NFI3. Red dots are *Pinus sylvestris* plots; yellow triangles are *Pinus nigra* plots; blue squares are *Pinus pinaster* plots and other in green color are NFI3 plots not filling requirements for this study. Palencia province, Spain, 2003.

### 3.2 Field Data

For any plot of coniferous in the study area, data from Spanish NFI3 have been used. After the typical plot unit is a circumference of 15m of radius, an extrapolation to achieve the  $\text{Mg}\cdot\text{ha}^{-1}$  have been applied. Criteria for the selection of the plots was as follows:

- a) plots with main species *Pinus sylvestris* L. (Valsaín pine, red pine), *Pinus nigra* Arn. (black pine) homogeneous in their composition,
- b) plots with larger trees, with a normal diameter equal to or greater than 7.5 cm,
- c) plots with a fraction cover (FCC) greater than or equal to 80% for *Pinus sylvestris* and *Pinus nigra* stands. These values of FCC for both species ensure the observation of a homogeneous stand and the continuity of the tree canopy without distorting the spectral response,
- d) plots with shrub species less than 50 cm high, which is equivalent to a negligible understory biomass for the spectral response.

### 3.3. Allometric Equations

Allometric equations for *Pinus pinaster* Ait. and *Pinus nigra* Arn. have been applied over the data included in any plot of the NFI3 included in the study area [13]. By using parameters of height and diameter of any individual, total biomass of dry matter has been achieved.

### 3.3 Satellite Imagery

To keep the temporal consistence between the field data of NFI3 and satellite imagery, Landsat-5 images have been used (Table 1). The imagery has been downloaded at a L1T processing level and was atmospherically corrected until a reflectance at a bottom of the atmosphere by DOS-method [14] [15] [16].

**Table 1.** Landsat-5 imagery data

Image/Level	Path/Row	Acquisition date	Azimuth angle	Elevation angle
L5TM/L1T	202/30	2003/09/15	145.67098075	44.74514056

### 3.4 Data Analysis

Gaussian process regression (GPR) models are non-parametric kernel-based probabilistic models. Meanwhile linear regressions estimate the error from the database itself, a GPR generates a response from an interval of variables of the training data and a new vector as an input by introducing variables from a Gaussian process.

The main assumption is that a set of random variables at any finite combination of them are distributed along the Gaussian curve and then any number of observed variables will be distributed as Gaussian as well. Consequently, a GPR model is a probabilistic method and thus, it is possible to predict the outcome intervals from training models. Moreover, the results are not fitting a line of responses but lie over a probability interval [12].

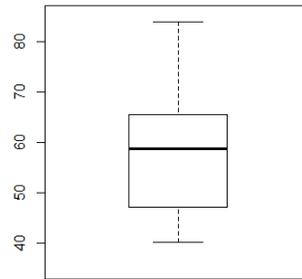
The covariance function between input and output variables shows the similarity between them. This kernel function and its mean (adjusted to zero for simplify) define the GPR [17]. In this study, a kernel function with separated length scale for each predictor is applied over the data, where an automatic relevance determination (ARD) method is used to order the inputs according to their importance [18].

GPR is a very suitable method for remote sensing analysis as it is not limited by the large number of parameters needed for the implementation of methods like neural networks [19] and its computation requirements are less demanding than those one based on pixel-by-pixel inversion methods [8]. However, GPR has not been tested yet on areas requiring a huge effort of inventory to simplify the field data.

## 4 Results

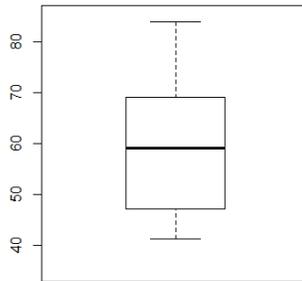
### 4.1 Inventory and Database

After choosing the plot filling the requirements and applying the allometric equations to calculate the biomass those values of the biomass were normalized to units concerning megagrams per hectare. Three databases have been used for the experiments. The first one includes all the values of pines without species segmentation (Fig. 2)

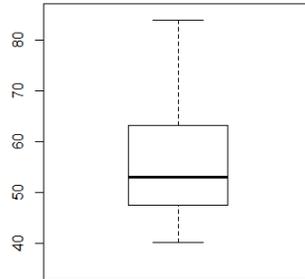


**Fig. 2.** Biomass sampling of a non-segmented database of pinus in Palencia ( $\text{Mg}\cdot\text{ha}^{-1}$ ), 2003.

In addition, two databases concerning individuals by species have been created for both *Pinus sylvestris* L. (Fig. 3) and *Pinus nigra* Arn. (Fig. 4).



**Fig. 3.** Biomass sampling of *Pinus sylvestris* L. in Palencia ( $\text{Mg}\cdot\text{ha}^{-1}$ ), 2003.



**Fig. 4.** Biomass sampling of *Pinus nigra* Arn. in Palencia ( $\text{Mg}\cdot\text{ha}^{-1}$ ), 2003.

For all plots ranged between 80 and 95% of fractional cover and for the biomass sampling without segmentation (Table 2), the mean ( $58.73 \text{ Mg}\cdot\text{ha}^{-1}$ ) is almost similar to the median value ( $58.72 \text{ Mg}\cdot\text{ha}^{-1}$ ). The difference between the median and the third quartile ( $6.64 \text{ Mg}\cdot\text{ha}^{-1}$ ) was smaller than that between the median and the first quartile ( $11.57 \text{ Mg}\cdot\text{ha}^{-1}$ ). As a result, the database displayed a slightly negative skewed distribution but was close to a normal one. The range of  $40.22$  to  $83.91 \text{ Mg}\cdot\text{ha}^{-1}$  is a normal value for pinus specimen of the same age. In addition, remarkable differences could not be found via a visual inspection. However, even with this asymmetry in the distribution of the biomass, these values and ranges were also reported by authors [12]; consequently, the sampling could be accepted as adequate for this study.

**Table 2.** Statistics of the biomass values for the pine plots without species segmentation ( $\text{Mg}\cdot\text{ha}^{-1}$ ) for the Palencia study area. 2003.

Minimum	1st quartile	Median	Mean	3rd Quartile	Maximum
40.22	47.15	58.72	58.73	65.36	83.91

If biomass sampling is segmented by species, for those 33 plots of *Pinus sylvestris* L. it is remarkable that mean ( $59.80 \text{ Mg}\cdot\text{ha}^{-1}$ ) and median ( $59.01 \text{ Mg}\cdot\text{ha}^{-1}$ ) are quite similar (Table 3). The difference between the median and the third quartile ( $9.96 \text{ Mg}\cdot\text{ha}^{-1}$ ) is now closer to that between the median and the first quartile ( $11.86 \text{ Mg}\cdot\text{ha}^{-1}$ ) with an almost normal distribution.

**Table 3.** Statistics of the biomass values of *Pinus sylvestris* L. ( $\text{Mg}\cdot\text{ha}^{-1}$ ) for the Palencia study area. 2003.

Minimum	1st quartile	Median	Mean	3rd Quartile	Maximum
41.25	47.15	59.01	59.80	68.97	83.76

Finally, for 15 plots of *Pinus nigra* Arn. it is noticeable that mean ( $56.38 \text{ Mg}\cdot\text{ha}^{-1}$ ) is higher than median ( $53.03 \text{ Mg}\cdot\text{ha}^{-1}$ ) (Table 4) and the difference between the median and the third quartile ( $10.12 \text{ Mg}\cdot\text{ha}^{-1}$ ) was higher than that between the median and the first quartile ( $5.44 \text{ Mg}\cdot\text{ha}^{-1}$ ). As a result, the database displayed a slightly positive skewed distribution but was close to a normal one.

**Table 4.** Statistics of the biomass values of *Pinus nigra* Arn. ( $\text{Mg}\cdot\text{ha}^{-1}$ ) for the Palencia study area. 2003.

Minimum	1st quartile	Median	Mean	3rd Quartile	Maximum
40.22	47.59	53.03	56.38	63.15	83.91

It is important to remark that 15 plot of *Pinus nigra* Arn. have influenced the values of those plots of *Pinus sylvestris* L. in a non-segmented database and, consequently, this species will be under-represented in a global assessment.

## 4.2 Biomass Assessment

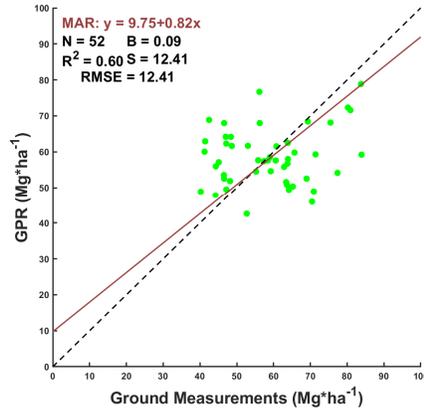
GPR were trained with Landsat-5 spectral bands 1 to 4 (450 to 900 nm), both included, and a combination of SWIR bands 5 (1,650 nm) and 7 (2,215 nm) as a summa of both as inputs, whereas total biomass represents the output. Any additional synthetic band is created because non-parametric methods can extract all the relevant information of the bands without user intervention. Other four non-forested random pixels have been included in the database to allow GPR to learn about land use classes [20]. Field data was randomly split in six subsets with well-distributed values of biomass.

Moreover, four subsets were used for algorithm training in each iteration and one subset was utilized for the result validation. The aim of the iterations is the generation of every possible combination between training and validating subsets to make the validation more robust by using all the subsets for this purpose. The performance has been evaluated with the absolute mean square root error (RMSE) and the coefficient of determination ( $R^2$ ) as overall indicators of accuracy.

A unique generic algorithm has been obtained using all the ground data points to understand the performance of GPR over pine structures. Afterwards, split database according to species have been created. To understand the fit between observed and predicted values, a major axis regression (MAR) line corresponding to the well-adjusted slope has been included [21].

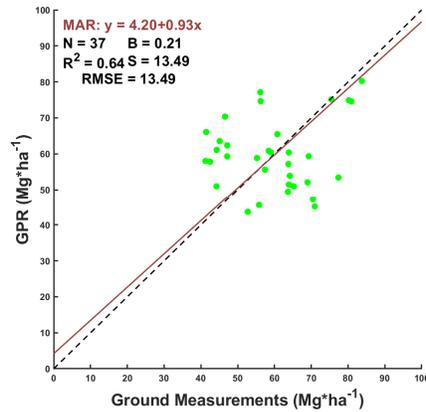
Figure 5 shows the scatterplot between the best-fitted model obtained by GPR and biomass ground data for global database including all the species. GPR have learnt about the difference between forest and non-forest data. Using all these points for training a GPR with Landsat-5 data, the  $R^2$  between those estimated values against field data could account for up to 60% of the variability of the forest.

The error, evaluated as root-mean-square error (RMSE), accounted for  $12.41 \text{ Mg}\cdot\text{ha}^{-1}$  in the gathered biomass range of 40.22 and  $83.91 \text{ Mg}\cdot\text{ha}^{-1}$ , representing a relative error between 14.7% and 30.8 % in the biomass assessment. In some cases, biomass was underestimated, whereas, in others, it was overestimated.



**Fig. 5.** Measured vs. estimated (GPR) biomass values for Landsat-5 and a database without species segmentation MAR.: major axis regression fit; N: number of samples; B: mean bias; R<sup>2</sup>: coefficient of determination; S: standard deviation of the bias; RMSE: root-mean-square error. The dashed line corresponds to the 1:1 line, while the continuous line is the MAR fit. Green points are generic forest plots.

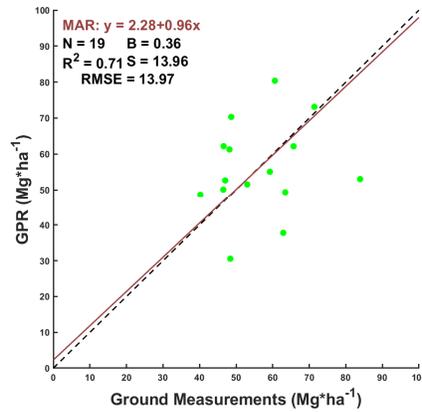
In the next step, biomass classes have been split to correspond the reflectance values according to their features. Figure 6 shows the scatterplot between the best-fitted model obtained by GPR and biomass ground data for *Pinus sylvestris* L. database.



**Fig. 6.** Measured vs. estimated (GPR) biomass values for Landsat-5 and a database of *Pinus sylvestris* L. MAR.: major axis regression fit; N: number of samples; B: mean bias; R<sup>2</sup>: coefficient of determination; S: standard deviation of the bias; RMSE: root-mean-square error. The dashed line corresponds to the 1:1 line, while the continuous line is the MAR fit. Green points are *Pinus sylvestris* L. forest plots.

The R2 account in this experiment reach up to 64% of the variability of the specific forest although the error, evaluated as root-mean-square error (RMSE), accounted for 13.49 Mg·ha<sup>-1</sup> in the gathered biomass range of 41.25 and 83.76 Mg·ha<sup>-1</sup>, representing a relative error between 16.1% and 32.7 % in the biomass assessment.

Concerning *Pinus nigra* Arn., figure 7 shows the scatterplot between the best-fitted model obtained by GPR and biomass ground data for *Pinus nigra* Arn. database. The R<sup>2</sup> account in this experiment reach in that case up to 71% of the variability of the specific forest although the error, evaluated as root-mean-square error (RMSE), accounted for 13.49 Mg·ha<sup>-1</sup> in the gathered biomass range of 40.22 and 83.91 Mg·ha<sup>-1</sup>, representing a relative error between 16.6% and 34.7 % in the biomass assessment.



**Fig. 7.** Measured vs. estimated (GPR) biomass values for Landsat-5 and a database of *Pinus nigra* Arn. MAR.: major axis regression fit; N: number of samples; B: mean bias; R<sup>2</sup>: coefficient of determination; S: standard deviation of the bias; RMSE: root-mean-square error. The dashed line corresponds to the 1:1 line, while the continuous line is the MAR fit. Green points are *Pinus nigra* Arn. forest plots.

Concerning to the linear MAR fit between predictions and ground values, the assessment for *Pinus nigra* Arn. revealed the most fitted slopes (0.71) and the smaller offset where the y-intercept was roughly 2.28 Mg·ha<sup>-1</sup>, a value that shows a barely uncovered terrain (Table 5). All these results indicate that the combined use of random field and low-intensity sampling together with a minimum segmentation of the database ensured by a normal distribution, could provide satisfactory results for forest management purposes by using probabilistic models applied to remote sensing imagery of Landsat-5.

**Table 5.** Statistics of the performance of a Gaussian process depending on the story segmentation. MAR: major axis regression fit; R<sup>2</sup>: coefficient of determination; RMSE: root-mean-square error.

Species	MAR	R <sup>2</sup>	RMSE
Unsegmented	$y = 9.75 + 0.82 \cdot x$	0.60	12.41
<i>Pinus sylvestris</i> L.	$y = 4.20 + 0.93 \cdot x$	0.64	13.49
<i>Pinus nigra</i> Arn.	$y = 2.28 + 0.96 \cdot x$	0.71	13.97

## 5 Discussion

The study focuses on the province of Palencia and in two types of pine forest. With these parameters, a cost-effective and affordable methodology for assessing biomass stock that requires a very low number of forest inventories has been proposed. This methodology applies Gaussian process regression methods over Landsat-5 surface reflectance and forest inventories data obtained after fieldwork.

This research allows to apply an operative method of machine-learning, a non-parametric empirically based method of Gaussian process regressions for biomass assessment at decametric resolution for Landsat-5. This method has been previously demonstrated to be an excellent method in terms of cost-effectiveness in a temperate oak coppice forests with shrubs [20].

Other methodologies have been purposed for accurate forest inventories, by traditional or even LiDAR means [22] but the methodology here reproduced reduces the cost and is able to represent up to 71% of the variability of the forest with only one plot each ten hectares, which is the typical distribution of the NFI [3].

This regression method needs no explicit selection of spectral bands and spectral resolutions of the TM sensor of Landsat-5 are valid for this purpose. It means that an historical assessment can be done with this methodology, as it overpasses the results obtained by traditional regressions between biophysical values and optical indexes. Avoiding the user intervention making an aprioristic selection of the bands allows the GPR to optimize the extraction of the relevant information of each band.

The biomass range in this study lies between 40 and 84 Mg·ha<sup>-1</sup> and the RMSE obtained is low, that shows a high level of determination found. Indeed, an average error of around 20% of total biomass in this kind of forests has been obtained, in accordance with previous experiments [20].

Although some spectral methods for land use and land classification have been proposed for a segmentation of the areas [23], it is recommended the use of ground-truth maps to apply algorithms over the same forest structures. According to [24,25] segmentation of the classes is necessary to avoid errors in the biomass assessment, as reflectance values are not valid to segment forest structures in narrow areas.

## 6 Conclusions

As a conclusion, the combination of Landsat-5 imagery, Gaussian process regressions, a reduced database and a story segmentation can be considered as a suitable method for pine forest biomass assessment and can be an essential tool for affordable and adaptive forest inventories and management. Finally, as a major advantage of this method, it has been demonstrated that a small database can deliver excellent results and avoid long and expensive forest inventories needed for forest management at both local and regional scale.

## References

1. Navarrete-Poyatos, M.A.; Navarro-Cerrillo, R.M.; Lara-Gómez, M.A.; Duque-Lazo, J.; Varo, M.d.l.A.; Palacios Rodriguez, G. Assessment of the Carbon Stock in Pine Plantations in Southern Spain through ALS Data and K-Nearest Neighbor Algorithm Based Models. *Geosciences* 2019, 9, 442. <https://doi.org/10.3390/geosciences9100442>.
2. González-Díaz, P.; Ruiz-Benito, P.; Gosálbez Ruiz, J.; Chamorro, G.; Zavala, M.A. A Multifactorial Approach to Value Supporting Ecosystem Services in Spanish Forests and Its Implications in a Warming World. *Sustainability* 2019, 11, 358. <https://doi.org/10.3390/su11020358>.
3. DGCN. III Inventario Forestal Nacional Español 1996–2006; Dirección General de Conservación de la Naturaleza, Ed.; Ministerio de Medio Ambiente: Madrid, Spain, 2006.
4. Aranha, J.; Enes, T.; Calvão, A.; Viana, H. Shrub Biomass Estimates in Former Burnt Areas Using Sentinel 2 Images Processing and Classification. *Forests* 2020, 11, 555. <https://doi.org/10.3390/f11050555>.
5. Dong, J.; Kaufmann, R.K.; Myneni, R.B.; Tucker, C.J.; Kauppi, P.; Liski, J.; Buermann, W.; Alexeyev, V.; Hughes, M.K. Remote sensing estimates of boreal and temperate forest woody biomass: Carbon pools, sources, and sinks. *Remote Sens. Environ.* 2003, 84, 393–410. <https://doi.org/10.1016/S0034-425700130-X>.
6. Muukkonen, P.; Heiskanen, J. Estimating biomass for boreal forest using ASTER satellite data combined with standwise forest inventory data. *Remote Sens. Environ.* 2005, 99, 464–447. <https://doi.org/10.1016/j.rse.2005.09.011>.
7. Kumar, L.; Sinha, P.; Taylor, S.; Alqurashi, A.F. Review of the use of remote sensing for biomass estimation to support renewable energy generation. *J. Appl. Remote Sens.* 2015, 9, 097696. <https://doi.org/10.1117/1.JRS.9.097696>.
8. Verrelst, J.; Camps-Valls, G.; Muñoz-Marí, J.; Rivera, J.P.; Veroustraete, F.; Clevers, J.G.P.W.; Moreno, J.J. Optical remote sensing and the retrieval of terrestrial vegetation biogeophysical properties—A review. *ISPRS J. Photogramm. Remote Sens.* 2015, 10, 273–290. <https://doi.org/10.1016/j.isprsjprs.2015.05.005>.
9. Combal, B.; Baret, F.; Weiss, M.; Trubuil, A.; Macé, D.; Pragnère, A.; Myneni, R.; Knyazikhin, Y.; Wang, L. Retrieval of canopy biophysical variables from bi-directional reflectance. Using prior information to solve the ill-posed inverse problem. *Remote Sens. Environ.* 2003, 84, 1–15. <https://doi.org/10.1016/S0034-425700035-4>.
10. Verrelst, J.; Alonso, L.; Camps-Valls, G.; Delegido, J.; Moreno, J. Retrieval of vegetation biophysical parameters using Gaussian process techniques. *IEEE Trans. Geosci. Remote Sens.* 2012, 50, 1832–1843. <https://doi.org/10.1109/TGRS.2011.2168962.4>.
11. Wang, T.; Xiao, Z.; Liu, Z. Performance evaluation of machine learning methods for leaf area index retrieval from time-series MODIS reflectance data. *Sensors* 2017, 17, 81. <https://doi.org/10.3390/s17010081>.

12. Rasmussen, C.E. Gaussian Processes in Machine Learning. In *Advanced Lectures on Machine Learning; Lecture Notes in Computer Science*, Bousquet, O., von Luxburg, U., Rätsch, G., Eds.; Springer: Berlin/Heidelberg, Germany, 2004; Volume 3176. [https://doi.org/10.1007/978-3-540-28650-9\\_4](https://doi.org/10.1007/978-3-540-28650-9_4).
13. Ruiz-Peinado, R., del Río, M., Montero, G., 2011: New models for estimating the carbon sink capacity of Spanish softwood species. *Forest Systems* 20: 176-188. DOI: 10.5424/fs/2011201-11643.
14. Hantson, S., Chuvieco, E., Pons, X., Domingo, C., Cea, C., More, G., Cristobal, J., Peces, J.J., Tejero, J.A. (2011). Cadena de pre-procesamiento estándar para las imágenes Landsat del Plan Nacional de Teledetección. *Revista de Teledetección* 36, 51-61.
15. Chavez, P.S. (1988). An improved dark-object subtraction technique for atmospheric scattering correction of multispectral data. *Remote Sensing of Environment* 24(3): 459-479.
16. Chavez, P.S. (1996). Image-based atmospheric corrections. Revisited and improved. *Photogrammetric Engineering and Remote Sensing* 62(9): 1025-1036.
17. Camps-Valls, L.; Gómez-Chova, J.; Muñoz-Marí, J.; Vila-Francés, J.; Amorós, J.; del Valle-Tascon, S.; Calpe-Maravilla, J. Biophysical parameter estimation with adaptive Gaussian Processes. In *Proceedings of the Geoscience and Remote Sensing Symposium, University of Cape Town, Cape Town, South Africa, 2009; Volume 4*.
18. Neal, R.M. *Bayesian Learning for Neural Networks*; Springer: New York, NY, USA, 1996.
19. MacKay, D.J.C. Introduction to Gaussian processes. In *Neural Networks and Machine Learning*; Bishop, C.M., Ed.; Kluwer Academic Press: Dordrecht, The Netherlands, 1998; pp. 133–166.
20. Vinué-Visús, D.; Ruiz-Peinado, R.; Fuente, D.; Oliver-Villanueva, J.-V.; Coll-Aliaga, E.; Lerma-Arce, V. Biomass Assessment and Carbon Sequestration in Post-Fire Shrublands by Means of Sentinel-2 and Gaussian Processes. *Forests* 2022, 13, 771. <https://doi.org/10.3390/f13050771>
21. Harper, W.V. Reduced major axis regression: Teaching alternatives to least squares. In *sustainability in statistics education. In Proceedings of the Ninth International Conference on Teaching Statistics (ICOTS9), Flagstaff, AZ, USA, 13–18 July 2014; Makar, K., De Sousa, B., Gould, R., Eds.; International Statistical Institute: Voorburg, The Netherlands, 2014; pp. 1–4*.
22. Vinué, D.; Camacho, F.; Fuster, B. Above-ground biomass and biophysical variables estimation in Mediterranean forest using terrestrial laser scanner. A case of study in Liria (Spain). In *Fifth Recent Advances in Quantitative Remote Sensing*; Sobrino, J.A., Ed.; Universitat de València: Valencia, Spain, 2017.
23. Talucci, A.C.; Forbath, E.; Kropp, H.; Alexander, H.D.; DeMarco, J.; Paulson, A.K.; Zimov, N.S.; Zimov, S.; Loranty, M.M. Evaluating Post-Fire Vegetation Recovery in Cajander Larch Forests in Northeastern Siberia Using UAV Derived Vegetation Indices. *Remote Sens.* 2020, 12, 2970. <https://doi.org/10.3390/rs12182970>
24. Kumar, L.; Sinha, P.; Taylor, S.; Alqurashi, A.F. Review of the use of remote sensing for biomass estimation to support renewable energy generation. *J. Appl. Remote Sens.* 2015, 9, 097696. <https://doi.org/10.1117/1.JRS.9.097696>
25. Galidaki, G.; Zianis, D.; Gitas, I.; Radoglou, K.; Karathanassi, V.; Tsakiri-Strati, M.; Woodhouse, I.; Mallinis, G. Vegetation biomass estimation with remote sensing: Focus on forest and other wooded land over the Mediterranean ecosystem. *Int. J. Remote Sens.* 2017, 38, 1940–1966. <https://doi.org/10.1080/01431161.2016.1266113>

# Development of a new AI-based methodology to assess working postures

Sonia Serna Arnau<sup>1</sup>, Sabina Asensio-Cuesta<sup>2</sup>, Juan Carlos González García<sup>1</sup>, and Enrique Medina Ripoll<sup>1</sup>

<sup>1</sup> Instituto de Biomecánica de Valencia (IBV), Universitat Politècnica de València, Valencia, Spain.

<sup>2</sup> Instituto de Tecnologías de la Información y Comunicaciones (ITACA), Universitat Politècnica de València, Valencia, Spain

**Abstract.** This paper aims to provide a comprehensive description of the ongoing procedure undertaken for the development of an innovative methodology based on artificial intelligence (AI). The proposed methodology seeks to enhance the efficiency of ergonomic assessment regarding postural load in comparison to existing practices. By leveraging AI, this research tries to overcome the limitations associated with traditional approaches and looks for offering a more accurate and effective solution. Additionally, this paper presents the current stage of development achieved thus far. Furthermore, it discusses certain limitations and explores potential avenues for further advancements and improvements in the system development process.

## 1 Introduction

Maintaining awkward postures implies the need for greater muscular effort since muscles cannot work efficiently if they are in extreme range of motion positions. There is evidence that links adopting awkward working postures with less performance efficiency and an increased risk of injuries or musculoskeletal disorders (MSD) [1-8].

MSDs are a very serious problem which generates, in addition to health problems, work absenteeism, huge economic losses and reduced productivity [8].

Therefore, it is very important to manage and prevent MSDs, for which it is necessary to apply an adequate ergonomic assessment method to analyze postural risks. Applying an adequate method will subsequently allow the development and implementation of the corresponding corrective measures, which reduce the workload to acceptable levels for workers [9].

There is a wide variety of validated methodologies for ergonomic postural analysis, such as RULA (Rapid Upper Limb Assessment) [10], REBA (Rapid Entire Body Assessment) [11], OWAS (Ovako Working Analysis System) [12], Keyserling Checklist [13], 3DSSPP- Chaffin [14], ALLA (Agricultural lower limb assessment) [15], Posture targeting [16], HARBO (Hands relative to the body) [17], LUBA (Postural loading on the upper-body assessment) [18], PATE Instrument (Patient Handling) [19]

or DINO (Direct Nurse Observation instrument for the assessment of work technique during patient transfers) [20].

Ergonomic postural assessment methodologies include OWAS, being one of the most frequently cited and used in numerous work sectors [8] [9] [21] [22]. Some of the primary advantages are: it is a simple and useful method that can be used by personnel from any field, even without specialized training [8] [12]; it is well documented and widely endorsed [8] and offers reliable and valid results, with high intra- and inter-rater reliability [9] [22]; and a high predictive validity which relates awkward working postures defined by OWAS with the appearance of MSD.

The reliability of the method has been shown greater the more positions are coded. A study with 20,601 observations investigated the effects of postural sampling interval deviation and concluded that, in the working postural analysis using OWAS, direct observations should have a sampling interval of 20 s (due to human limitations), and that video-based observations should have a sampling interval of 10 s or less, since with sampling intervals of 10 s or less, the experimental error is lower than the theoretical one, still acceptable for sampling intervals up to 60 seconds [22].

This is because, in many sectors, while carrying out a task, in a very short time (less than a second), the posture of various body segments can change significantly [23], assuming the need to split the video into a larger number of frames. This would greatly increase the analysis effort. This can hinder certain studies' scope due to the limited availability of time for its application [8], [22] [23], so the assessment time is one of the major drawbacks of the method.

There already exist systems such as computer software, and methods using range sensors (Kinect) [24] [25], 3D cameras, EMG sensors, and other new technologies allowing the collection of information to apply the OWAS method [8]. Studies are comparing the traditional assessment (without using software) with the computerized assessment, and concluding the operator's working time optimization [24].

Artificial intelligence (AI) refers to the ability of machines to perform tasks that would normally require human intervention and to make decisions based on data and patterns. Although AI is not a new term, since the research discipline of AI was founded at the Dartmouth Conference in New Hampshire in 1956 [25], this technology has seen a rapid advancement in recent years and is used in a wide variety of fields, such as medicine, industry, security and research.

There are numerous studies and tools about employing artificial intelligence (AI) to facilitate ergonomic analysis. A systematic review, analyze 25 relevant studies, focused on the combined utilization of wearable devices and AI for ergonomic purposes. The review revealed a notable surge in interest in recent years, particularly in monitoring workers' biomechanical risks and exploring the latest applications of AI and wearable sensors in the domain of physical ergonomics [26], ensuring that this incorporation of AI is becoming increasingly prevalent within the field of ergonomics.

In addition to wearables, another AI-based method employed for supporting ergonomic analysis is the utilization of motion capture (MOCAP) systems. These systems utilize AI to assist ergonomic experts in posture detection. For instance, a study proposes a combination of a neural network, specifically Convolutional Pose Machines, with the Rapid Entire Body Assessment (REBA) method to establish an automated risk assessment system [27]. Another investigation explores the benefits of employing

AI and MOCAP systems in tandem to enhance ergonomic observational methods [28]. Several algorithms are trained in this study, with the aim of providing guidelines for AI-assisted ergonomic assessment based on the observation of multiple workers and it concludes that such approaches have the potential to be effective in aiding the computation of ergonomic assessments.

Achieving, from AI, the complete automation of data collection to apply the ergonomic assessment methodologies, would be very useful since it would allow the possibility of analyzing large amounts of data (images extracted from frames video) and calculating the level of risk of suffering from MSD more reliably. In addition, since OWAS is a simple methodology, with high intra- and inter-rater reliability, it will be easier to analyze the validity of the new AI-based methodology [22] [29].

This paper aims to design the procedure to follow to achieve an improved AI-based methodology which would assess working postures load, based on a previous methodology developed by the IBV [30]. The new methodology endeavors to improve precision, simplicity, and time efficiency, while additionally exploring the feasibility of incorporating the variable of sex. Furthermore, this paper seeks to provide an overview of the current progress achieved in the developmental process thus far.

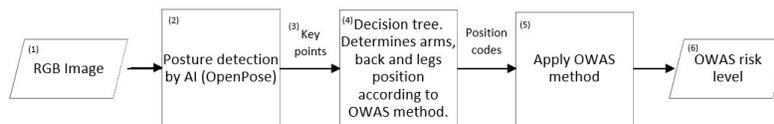
## 2 Methodology

The main objective of this paper is to develop software that is based on artificial intelligence (AI) capabilities to enable the automation of postural analysis using the OWAS method. Furthermore, we aim to investigate the viability and potential benefits of incorporating additional factors, such as sex, which may contribute to variations in the risk of developing musculoskeletal disorders (MSDs) [21].

### 2.1 Current model structure

Starting from previous methodology (ErgoIA), accessible via the website (<https://ergoia.ibv.org>), this study uses artificial intelligence (AI) techniques to facilitate the automated analysis of individuals' work-related movements, thereby simplifying the process of ergonomic assessment.

ErgoIA adheres to the following postural risk assessment process based on OWAS [31] (Fig. 1):



*Fig. 1. ErgoIA process to obtain OWAS risk level*

The detailed process phases are outlined below:

1.- The initial phase involves input data entry, consisting of RGB images extracted from videos.

2.- Upon loading the videos into the software, they are transmitted to a server for processing through the application of the OpenPose neural network, employing the MobileNet v1 architecture as a feature extractor [32], with additional post-processing layers incorporated to calculate parameters essential for ergonomic assessment.

3.- The neural network identifies 17 key points pertaining to the facial, trunk, arm, and leg regions, thereby determining the person's posture in the image.

4.- Subsequently, a decision tree, functioning as a complementary component to the neural network, facilitates the extraction of parameters necessary to ascertain the positioning of body segments in accordance with the OWAS postural coding scheme. This coding scheme encompasses five potential back positions, three potential arm positions, and seven potential leg positions (Figure 2).

5.- This step leads to the codification of the posture, which corresponds to the input data required for the application of the OWAS method. By default, the load code is set to 1 (indicating loads less than 10 kg), as the OpenPose system cannot detect the specific load magnitude.

6.- The output data generated entails the assessment of posture risk levels, considering the load as less than 10 kg. Software users have the option to review and rectify the coded postures, as well as manually assign a different load code (2 for loads ranging from 10 to 20 kg, and 3 for loads exceeding 20 kg) when necessary.

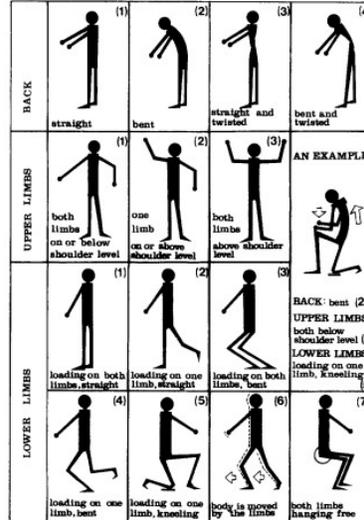
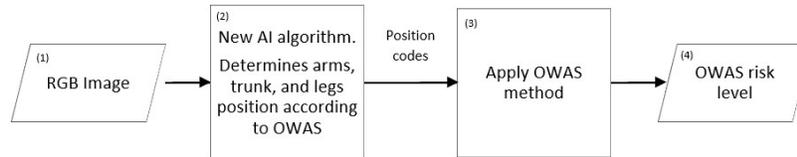


Fig. 2. List of the possible back, upper limbs and lower limbs positions according to OWAS classification and an example of posture description. [12]

## 2.2 Structure of the future model to assess postural load model based on OWAS

Starting from this methodology, ErgoIA seeks to simplify the process to reach the following one (Fig. 3).



**Fig. 1.** New AI-based methodology process to obtain OWAS risk level

Here we describe the future model steps:

- 1.- The process begins with the input data entry, comprising RGB images extracted from videos.
- 2.- Subsequently, a new algorithm is employed to directly extract the posture codification from the image, adhering to the OWAS coding scheme. By default, the load code is set to 1 (indicating loads less than 10 kg).
- 3.- The OWAS methodology is then implemented, using the posture codification obtained from the algorithm.
- 4.- The resulting output data encompasses the OWAS posture risk levels, assuming a load code of less than 10 kg. Users of the software have the option to review and rectify the coded postures, and manually assign a different load code (2 for loads ranging from 10 to 20 kg, and 3 for loads exceeding 20 kg) if necessary.

## 2.3 Description of the process necessary to achieve the future model of postural load based on OWAS

To achieve the development of a new posture detection model, a process comprising five distinct steps or phases is proposed. Each of these steps, crucial to the construction of the novel model, is described in detail below.

- 1.- Data collection: RGB images are collected for training the AI model. The availability of a sizable dataset is essential to ensure accurate and meaningful predictions from AI models [33]. Data collection is performed using ErgoIA, where users with different profiles, including client users, students attending ErgoIA training, and software developers, upload videos. Prior to utilizing the images from these videos, individuals appearing in each video are required to provide informed consent for the processing of their images for research, development, and innovation purposes in the field of human movement analysis conducted by the Instituto de Biomecánica de Valencia (IBV).

2.- Data labelling: Experts with advanced ergonomics training under-take the task of labelling the collected data. Initially, the existing method (ErgoIA, Fig.3) is applied, followed by subsequent corrections, including:

- Rectifying positions that the AI model may have failed to code accurately.
- Eliminating frames that do not meet the necessary conditions for postural analysis.

Participating experts are IBV researchers who can demonstrate a profound understanding and significant expertise in the realm of ergonomics. These experts possess the necessary qualifications to practice as "Técnico Superior en Prevención de Riesgos Laborales" (Senior Technician in Occupational Risk Prevention) in Spain, specifically with at least the specialization in ergonomics and psychosociology (Ergonomía y Psicología). Furthermore, they actively participate in the fields of ergonomics and occupational health in their usual work environment.



**Fig. 3.** Screenshot of ErgoIA, an example of a posture OWAS coding.

The essential conditions for postural analysis entail ensuring that the full body of the individual in the frame is visible without obstructions, and distinguishing the body segments (arms, trunk, and legs) coded by the OWAS method. Additionally, only a single person should be present in the analyzed frame, and the image quality should be sufficient to distinguish the person, avoiding issues such as blurriness, motion blur, low contrast, or being out-of-focus.

In addition, the sex of the individuals depicted in the images will be annotated to facilitate subsequent analysis aimed at investigating potential correlations between the level of postural risk, the adopted postures during task performance, and the sex of the workers. The incorporation of this variable in the new model will be studied depending on the results obtained.

3.- Selection of a pre-trained neural network: A pre-trained neural network specialized in person and posture detection will be chosen to reduce the amount of data required for network training. The neural network will be chosen based on the amount of data that can be collected.

4.- Neural network training: The collected and labelled data are used to train the selected neural network. This training enables the network to establish associations between detected postures and the 84 possible combinations of postures (arms, trunk, and legs) outlined by the OWAS methodology. By default, a load code of 1 (indicating loads less than 10 kg) is considered. Consequently, the corresponding level of risk associated with each posture is obtained. If higher loads are present, the load code can be manually adjusted to 2 (10-20 kg) or 3 (greater than 20 kg), leading to the determination of the corresponding risk level. Notably, the incorporation of the sex variable into the ergonomic analysis is pursued, aiming to investigate potential associations between sex and the OWAS risk level.

5.- Experts validation process: The trained neural network undergoes a validation process conducted by experts. The validation primarily focuses on verifying whether the new network accurately encodes postures according to the OWAS methodology, considering the positions of the arms, trunk, and legs, while disregarding the load factor. Additionally, the hypothesis that employing an AI-based methodology for postural analysis saves time will be examined. Ergonomics experts will perform various postural analyses using two approaches:

- The new neural network trained according to the procedure outlined in this paper.
- The OWAS module of Ergo/IBV Software [34], a software developed by the Instituto de Biomecánica de Valencia (IBV) for assessing ergonomic risks at work.

The analysis will consider the number of postures that the new network may have failed to code accurately, as well as the time required by the experts to perform the ergonomic analysis in both cases.

### 3 Results

In the ongoing process phase aimed at achieving the future model of postural load based on the OWAS methodology (as discussed in Section 2.3), the progress thus far encompasses steps 1 and 2. A summary of the analyzed data is presented in Table 1.

**Table 1.** Summary of data already analyzed, to 30/05/2023

Videos collected	Total frames analyzed	Discarded Frames (units and %)	Labelled Frames (units and %)
117	7520	2404 – 31,97%	5116 – 68,03%

Table 1 illustrates that to date (30/05/2023), a total of 117 videos have been gathered. From these videos, 7520 frames have been obtained (RGB images) and have been subjected to analysis. Within this dataset, 2404 frames have been excluded due to the presence of at least one of the following criteria: absence of a person, multiple persons, incomplete depiction of the body segments required for OWAS coding, or the frames being blurry or out-of-focus. Consequently, a total of 5116 valid frames have been identified and subsequently labelled.

This implies that approximately 31.97% of frames have been discarded, a considerable proportion.

## 4 Discussion

One plausible explanation for this substantial number of excluded frames (31.97% of total frames analyzed) may be attributed to the fact that many users and students, who are not yet proficient with the software, upload test videos without ensuring that all the necessary conditions for accurate processing are met.

It is also important to acknowledge that certain occupations inherently pose challenges in capturing recordings of a single individual or in an unobstructed view of all body segments. This limitation must be considered throughout the development of this methodology.

Another noteworthy consideration pertains to the involvement of experts in both the labelling process and the validation of the new model, specifically about steps 2 and 5 of the process aimed at achieving the future model (section 2.3). While these experts are knowledgeable in the same field, it is important to acknowledge that each researcher brings their own unique experiences and areas of expertise. Consequently, the potential for biases to emerge in their results should be carefully examined. It is imperative to explore approaches that can yield more dependable assessment outcomes, along with assessing intra- and inter-rater reliabilities.

Remarkably, the OWAS methodology boasts a high degree of intra-rater reliability [22] [29], being one of the reasons why this methodology has been used for the implementation of AI in this new model.

The ultimate objective is to persist in advancing this process to create an AI-based methodology that effectively evaluates the workload associated with different working postures. This approach seeks not only to save time for professionals but also incorporate the consideration of sex, where applicable.

## 5 Conclusions

The quantity of collected data indicates that a satisfactory sample will be acquired; however, there exists a limitation imposed by the proportion of discarded frames. The percentage of 31.97% is notably elevated, thereby necessitating the identification of approaches to mitigate this occurrence, thereby enhancing the efficiency of labelling. Additionally, an interesting subject of inquiry pertains to the inclusion of sex as a factor during the coding and analysis of frames. This investigation aims to explore potential associations between sex and postural risk and determine the feasibility of incorporating this variable within the methodology to proactively address musculoskeletal disorders (MSDs).

Despite the aforementioned limitations, it is anticipated that this research will contribute to the validation of AI implementation for ergonomic assessment. Furthermore, it will serve to support future investigations in this field, facilitating the development of more effective methodologies and reducing the time required for ergonomic analysis.

## References

1. Widanarko B, Legg S, Stevenson M, Devereux J, Eng A, 't Mannetje A, Cheng S, Pearce N. Gender differences in work-related risk factors associated with low back symptoms. *Ergonomics*. 2012;55(3):327-42. doi: 10.1080/00140139.2011.642410. PMID: 22409170.
2. Anita, A.R.; Yazdani, A.; Hayati, K.S.; Adon, M.Y. Association between awkward posture and musculoskeletal disorders (MSD) among assembly line workers in an automotive industry. *Malaysian J. Med. Health Sci.* 2014, 10, 23–28
3. NIOSH. (1997). Musculoskeletal disorders (MSD) and workplace factors. A critical review of epidemiologic evidence for work-related musculoskeletal disorders of the neck, upper extremity, and low back. Cincinnati: US Department of Health and Human Services. : Bernard B, Lawrence JF. Musculoskeletal Disorders and Workplace Factors -A Critical Review of Epidemiologic Evidence for Work-Related Musculoskeletal Disorders of the Neck, Upper Extremity, and Low Back. Cincinnati: Centers for Disease Control and Prevention (CDC), National Institute for Occupational Safety and Health (NIOSH); 1997.
4. Van Der Windt, D. A. W. M.. (2000). Occupational risk factors for shoulder pain: a systematic review. *Occupational and Environmental Medicine*, 57(7), 433–442. <https://doi.org/10.1136/oem.57.7.433>
5. Da Costa, B. R., & Vieira, E. R.. (2009). Risk factors for work-related musculoskeletal disorders: a systematic review of recent longitudinal studies. *American Journal of Industrial Medicine*, n/a–n/a. <https://doi.org/10.1002/ajim.20750>
6. Ayub Y, Shah ZA (2018) Assessment of Work Related Musculoskeletal Disorders in Manufacturing Industry. *J Ergonomics* 8: 233. doi: 10.4172/2165-7556.1000233
7. Kumar S. Theories of musculoskeletal injury causation. *Ergonomics*. 2001 Jan 15;44(1):17-47. doi: 10.1080/00140130120716. PMID: 11214897.
8. Gómez-Galán M, Pérez-Alonso J, Callejón-Ferre AJ, López-Martínez J. Musculoskeletal disorders: OWAS review. *Ind Health*. 2017 Aug 8;55(4):314-337. doi: 10.2486/indhealth.2016-0191. Epub 2017 May 9. PMID: 28484144; PMCID: PMC5546841.

9. Kee, D. Systematic Comparison of OWAS, RULA, and REBA Based on a Literature Review. *Int. J. Environ. Res. Public Health* 2022, 19, 595. <https://doi.org/10.3390/ijerph19010595>
10. McAtamney, L., and E. N. Corlett. 1993. "RULA: A Survey Method for the Investigation of Work-Related Upper Limb Disorders." *Applied Ergonomics* 24 (2): 91–99. doi:10.1016/0003-6870(93)90080-s.
11. Hignett, S., and L. McAtamney. 2000. "Rapid Entire Body Assessment (REBA)." *Applied ergonomics* 31 (2): 201–205. doi:10.1016/S0003-6870(99)00039-3
12. Karhu, O., P. Kansii, and I. Kuorinka. 1977. "Correcting Working Postures in Industry: A Practical Method for Analysis." *Applied ergonomics* 8 (4): 199–201. doi:10.1016/0003-6870(77)90164-8
13. Keyserling, W. M., M. Brouwer, and B. A. Silverstein. 1992. "A Checklist for Evaluating Ergonomic Risk Factors Resulting from Awkward Postures of the Legs, Trunk and Neck." *International Journal of Industrial Ergonomics* 9 (4): 283–301. doi:10.1016/0169-8141(92)90062-5
14. Chaffin, D. 1997. "Development of Computerized Human Static Strength Simulation Model for Job Design." *Human Factors and Ergonomics in Manufacturing* 7 (4): 305–322. doi:10.1002/(SICI)1520-6564(199723)7:43.0.CO;2-7.
15. Kong, Y.-K., J.-G. Han, and D.-M. Kim. 2010. "Development of an Ergonomic Checklist for the Investigation of WorkRelated Lower Limb Disorders in Farming – ALLA: Agricultural Lower-Limb Assessment." *Journal of the Ergonomics Society of Korea* 29 (6): 933–942. doi:10.5143/JESK.2010.29.6.933.
16. Corlett, E. N., S. J. Madeley, and I. Manenica. 1979. "I. Posture Targeting: A Technique for Recording Working Postures." *Ergonomics* 22 (3): 357–366. doi:10.1080/00140137908924619
17. Wiktorin, C., Mortimer, M., L. Ekenvall, A. Kilbom, and E. W. Hjelm 1995. "HARBO, a Simple Computer-Aided Observation Method for Recording Work Postures." *Scandinavian journal of Work, Environment & Health* 21 (6): 440–449. doi:10.5271/sjweh.60.
18. Kee, D., and W. Karwowski. 2001. "LUBA: An Assessment Technique for Postural Loading on the Upper Body Based on Joint Motion Discomfort and Maximum Holding Time." *Applied ergonomics* 32 (4): 357–366. doi:10.1016/s0003-6870(01)00006-0.
19. Kjellberg, K., C. Johnsson, K. Proper, E. Olsson, and M. Hagberg. 2000. "Imtion Instrument for Assessment of Work Technique in Patient Transfer Tasks." *Applied Ergonomics* 31 (2): 139–150. doi:10.1016/s0003-6870(99)00046-0.
20. Johnsson, C., K. Kjellberg, A. Kjellberg, and M. Lagerstrom. € 2004. "A Direct Observation Instrument for Assessment of Nurses' Patient Transfer Technique (DINO)." *Applied ergo-nomics* 35 (6): 591–601. doi:10.1016/j.apergo.2004.06.004.
21. Serna Arnau S, Asensio-Cuesta S, Porcar Seder R. Musculoskeletal disorders risk assessment methods: a scoping review from a sex perspective. *Ergonomics*. 2023 Jan 20:1-17. doi: 10.1080/00140139.2023.2168767. Epub ahead of print. PMID: 36636799.
22. Brandl, C., Mertens, A., & Schlick, C. M. (2017). Effect of sampling interval on the reliability of ergonomic analysis using the Ovako working posture analysing system (OWAS). *International Journal of Industrial Ergonomics*, 57, 68-73.
23. Justavino, F. C., Ramirez, R. J., Perez, N. M., & Borz, S. A. (2015). The use of OWAS in forest operations postural assessment: advantages and limitations. *Bulletin of the Transilvania University of Brasov. Series II: Forestry• Wood Industry• Agricultural Food Engineering*, 7-16.
24. Klippert J, Gudehus T, Zick J (2012) A software-based method for ergonomic posture assessment in automotive preproduction planning: Concordance and difference in using software and personal observation for assessments. *Human factors and ergonomics in manufacturing & service industries* 22, 156–75. (doi:10.1002/hfm.20370). [Google Scholar]

25. Dartmouth College. Summer research project on artificial intelligence (Vox of Dartmouth). Hanover: Dartmouth College; 1956.
26. Donisi L, Cesarelli G, Pisani N, Ponsiglione AM, Ricciardi C, Capodaglio E. Wearable Sensors and Artificial Intelligence for Physical Ergonomics: A Systematic Review of Literature. *Diagnostics (Basel)*. 2022 Dec 5;12(12):3048. doi: 10.3390/diagnostics12123048. PMID: 36553054; PMCID: PMC9776838.
27. Lee, Y. C., & Lee, C. H. (2022). SEE: A proactive strategy-centric and deep learning-based ergonomic risk assessment system for risky posture recognition. *Advanced Engineering Informatics*, 53, 101717.
28. Igelmo, V., Syberfeldt, A., Högberg, D., Rivera, F., & Luque, E. (2020). Aiding observational ergonomic evaluation methods using MOCAP systems supported by AI-based posture recognition. *Adv. Transdiscipl. Eng*, 11, 419-429.
29. Sanchis, M., de Rosario, H Medina, E. Pedrero, J.F., Parrilla, E., Gamón, A., Valls, A., Miralles, P.P. Evaluación de riesgos ergonómicos en el puesto de trabajo mediante visión artificial y redes neuronales. In: XX Congresso Brasileiro de Ergonomia - Virtual 2020. ABERGO, 2020. ISSN: 2358-5463
30. De Rosario, H., Medina-Ripoll, E., Pedrero-Sánchez, J. F., Sanchis-Almenara, M., Valls-Molist, A., & Miralles-Garcera, P. P. (2021). Ergonomic Assessment with a Convolutional Neural Network. A Case Study with OWAS. In *Proceedings of the 8th International Ergonomics Conference: ERGONOMICS 2020* 8 (pp. 65-71). Springer International Publishing.
31. Sanchis, M., de Rosario, H., Parrilla, E., Medina, E., Valls, A., Miralles, P., (2022) La Inteligencia Artificial aplicada a la evaluación de riesgos ergonómicos. *ORP Journal*. ISSN 2385-3832
32. Cao, Z., Simon, T., Wei, S. and Sheikh, Y. (2017). "Realtime Multi-person 2D Pose Estimation Using Part Affinity Fields," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 1302-1310, doi: 10.1109/CVPR.2017.143.
33. Igelmo, V., Syberfeldt, A., Högberg, D., Rivera, F., & Luque, E. (2020). Aiding observational ergonomic evaluation methods using MOCAP systems supported by AI-based posture recognition. *Adv. Transdiscipl. Eng*, 11, 419-429.
34. IBV (2000). Ergo/IBV – Evaluación de riesgos laborales asociados a la carga física. [García-Molina C., Chirivella C., Page A., Tortosa L., Ferreras A., Moraga R., Jorquera J.]. Instituto de Biomecánica de Valencia (IBV), Valencia.

# Comparison of the overheads provoked by the inclusion of different Error Correction Codes in Embedded Systems\*

J. Gracia-Morán<sup>1</sup>, L.J. Saiz-Adalid<sup>1</sup>, J.C. Baraza-Calvo<sup>1</sup>,  
D. Gil-Tomás<sup>1</sup>, and P.J. Gil-Vicente<sup>1</sup>

<sup>1</sup> Instituto ITACA, Universitat Politècnica de València  
Camino de Vera, s/n, 46022 Valencia, Spain  
{jgracia, ljsaiz, jcbaraza, dgil, pgil}@itaca.upv.es  
<https://gstf.blogs.upv.es/>

**Abstract.** During these last years, the use of embedded systems has grown exponentially, mainly due to the expansion of the Internet of Things (IoT). Data collected by IoT devices are sent to the cloud to be processed in datacenters. Edge Computing philosophy wants to change this “passive” behavior of IoT devices. The basic idea is to process data produced by IoT devices closer to where they were created, instead of sending them through long routes. New challenges have emerged with the change to the Edge Computing philosophy. One of them is reliability. IoT devices have been built with low-reliable components, reduced weight and volume, and not very high computing and memory capacity for low power consumption. With these conditions, how can we rely on the results obtained by these devices? In this work, we have tried to answer this question by analyzing the effects of the inclusion of different software-implemented Error Correction Codes in real embedded systems, typically used in IoT.

## 1 Introduction

The exponential growth of IoT devices has provoked an incredibly big amount of generated data [1]. Usually, the processing of these data is done in a different place. Data generated in IoT devices are sent through routers/gateways to be processed in remote units and stored in different and remote storages [2]. With the increase in the number of IoT devices (and the generated data), the necessary bandwidth multiplies, especially for time-sensitive applications [3].

This process can be speeded up if processing is done near data are generated. Edge Computing is a new computing paradigm that proposes the processing of the data produced on the edge of the network in this same edge. The main idea is to carry out data processing near the data source, saving latency time, bandwidth, power consumption, and even, cloud outages [4][5]. In addition, services can be provided at the edge of the network [3].

---

\* Grant PID2020-120271RB-I00 funded by MCIN/AEI/10.13039/501100011033.

Focusing on IoT devices, in traditional cloud computing, data produced are sent to the cloud, and data consumers send data requests to the cloud. As the amount of data transmitted is too large, this process will lead to huge unnecessary bandwidth and computing resource usage. Moreover, privacy protection requirements must be considered [5].

Nowadays, CMOS technology integration scale has allowed to design memory systems with a great storage capacity. However, this aggressive scaling has also caused an increase in memory failure rate [6]. As the memory cell critical charge and the energy needed to cause a Single Cell Upset (SCU) is also reduced, Multiple Cell Upsets (MCUs), that is, simultaneous errors in more than one memory cell, can also be induced by a single particle hit [7].

A possible solution is the use of Error Correction Codes (ECC) [8]. ECC are typically designed trying to minimize the number of redundant bits, as they are added to each word in the whole memory. In addition, ECC circuitries introduce some overheads, such as encoding and decoding latencies, silicon area, and power consumption.

Nevertheless, what happens in already built systems? That is, what happens when the memory of a system is already designed? And, what happens if this system is an IoT device? Can this type of device tolerate single or multiple faults?

If an IoT device is going to process data, we must rely on this processing. Thus, data should be protected. IoT devices have been built with low-reliable components, reduced weight and volume, and not very high computing and memory capacity for low power consumption. In this way, if we need fault tolerance, it must be implemented by software, as the low-cost hardware used doesn't have this protection. As far as we know, there exists one fault-tolerant microprocessor on the market that can be used for IoT devices [9]. However, few studies have been done analyzing the impact of the inclusion of an ECC in a non-fault tolerant IoT device, that are the most common IoT devices.

In this paper, we deep into this type of study, which began in [10][11]. YTo do this, we have developed a small meteorological monitoring system, in which we are interested in protecting different variables. This simple control system will help us to carry out the proposed study. Thus, we have included a whole series of ECC with different fault tolerance properties. Once added, we have measured the size of the ECC and checked if the code can be executed on time. In this way, we can examine the effect of including the ECC in the system.

This work is organized as follows. Section 2 describes the systems and ECC used. Section 3 explains the results obtained, and Section 4 concludes the paper.

## **2 System description**

### **2.1 Microprocessors used**

We have used two different systems to carry out our experiments. The first one is based in the STM32F429i-DISCOVERY card [12]. It includes a development kit with an ARM Cortex-M4 architecture. Some of the characteristics of this card are [12]:

- ARM-Cortex M4 architecture.
- Integrated debugging tool.
- 2.4'' QVGA TFT LCD.
- External 64 Mbit SDRAM.
- System clock: 180 MHz.
- A set of sensors and actuators (gyroscope, LEDs and buttons).

The second system is based on an Arduino UNO R3 board [13], with these characteristics:

- • ATmega328p microcontroller.
- • 32 KB Flash memory, and 2 KB SRAM, and 1KB EEPROM.
- • 16 MHz clock frequency.

In both systems, we have added a temperature and humidity sensor, as well as an RGB LED. Also, in the Arduino-based system, we have included an LCD1602 module (LCD screen with 16 columns and 2 rows).

Data is obtained by the of temperature and humidity sensor. This data is protected by the different ECC explained in the next subsection. RGB LED will activate different colors depending on the values of the data obtained by the temperature and humidity sensor. Also, these values will be shown in both LCDs. As the behavior of the system depends on values obtained from the temperature and humidity sensor, we have protected them with a series of ECC. The rest of variables of the system reinitialize each time they are used, so their protection is less critical. The idea is to protect and verify that the data obtained from the sensor is correct, without having to delay taking new measurements. In the future, we want to complete this control system by adding more sensors and actuators.

## 2.2 Error Corrections Codes used

Before describing ECC employed, we must remark that all these ECC have been designed and optimized for hardware implementations. In this work, we have adapted these implementations to functions that can be used by our software. In this way, we can study the influence of these ECC in the software size and execution time by implementing different encoding and decoding functions with diverse error coverages. Table 1 summarizes the fault tolerance of the different ECC.

### Hamming SEC code

Broadly used, Hamming SEC (Single Error Correction) code [21] is a linear block code that can correct 1-bit errors with the lowest redundancy, as well as simple and fast encoding and decoding operations. This ECC was one of the first ECC used to get data reliability. In this work we have used a Hamming (38, 32) SEC code.

### OLS SEC-DEC code

Orthogonal Latin Square (OLS) codes achieve multiple bit correction with simple and fast decoders. They are based on one step majority logic decoding (OS-MLD) [17][18][19]. OLS codes are derived from a set of mutually orthogonal latin squares.

Latin squares are based on an  $m \times m$  matrix with columns and rows built as permutations of integers, in such a way that each value appears once in each row and column. OLS codes present a simple and fast decoding implementation, although their redundancy is very high and not all word lengths fit in an OLS ECC directly, as no orthogonal matrices exist in some cases. In this work, we have used an OLS (55, 32) code able to correct 2 random bits in error.

**Table 1.** Redundancy and Fault Tolerance of each ECC

ECC	N° of data bits	N° of code bits	Fault Tolerance
<i>Hamming SEC</i>	32	6	<b>Correction:</b> single bit errors
<i>OLS SEC-DEC</i>	32	23	<b>Correction:</b> single and 2-bit random errors
<i>SEC-DED-8AEC</i>	32	32	<b>Correction:</b> 8-bit burst errors, 8-bit adjacent errors <b>Detection:</b> 2-bit random errors; 12-bit burst errors.
<i>SEC-DED-12AEC</i>	32	32	<b>Correction:</b> 8-bit burst errors, 12-bit adjacent errors <b>Detection:</b> 2-bit random errors; 12-bit burst errors.
<i>SEC-DED-16AEC</i>	32	32	<b>Correction:</b> 8-bit burst errors, 16-bit adjacent errors <b>Detection:</b> 2-bit random errors; 12-bit burst errors.
<i>SEC-DED-20AEC</i>	32	32	<b>Correction:</b> 8-bit burst errors, 20-bit adjacent errors <b>Detection:</b> 2-bit random errors; 12-bit burst errors.
<i>FUEC-DAEC</i>	16	7	<b>Correction:</b> single and 2-bit adjacent errors; <b>Detection:</b> 3- and 4-bit burst errors
<i>FUEC-TAEC</i>	16	8	<b>Correction:</b> single, 2-bit adjacent errors, 3-bit burst errors; <b>Detection:</b> 4-bit burst errors
<i>FUEC-QUAEC</i>	16	9	<b>Correction:</b> single, 2-bit adjacent errors, 3- and 4-bit burst errors;

### Ultrafast codes

Originally, the main objective of Ultrafast codes [14] implemented in hardware was the reduction of encoding and decoding latencies. The processing speed of their hardware implementations makes them especially suitable for systems in which speed is essential. Although a software implementation loses part of the advantages in latency reduction, the error coverage achieved by these codes makes them an interesting option for this work. In this way, although their redundancy in hardware is very high, in this work it can be acceptable, as we want to protect only a set of variables, not the complete memory.

The original ECC can correct single errors, up to 5 adjacent errors and detect 2 non-adjacent errors (summarized as SEC-5AEC-DED) on a 16-bit code word that encodes an 8-bit data word. From this ECC, it is possible to design various decoders with different error coverages (5AEC, 4AEC, etc.). Also, several encoders and decoders can be combined to protect longer data words. For instance, it is viable to generate ECC for data words of 16, 32 and 64 bits (and even longer data words) [13].

Thus, in this work, we have used next ECC (all of them designed to protect 32-bit data words): i) SEC-DED-8AEC; ii) SEC-DED-12AEC; iii) SEC-DED-16AEC; and iv) SEC-DED-20AEC.

All these ECC can correct up to 8-bit burst errors and detect 2 errors in random bits, and up to 12-bit burst errors. Also, each one of them can correct the adjacent errors pointed (8, 12, 16, and 20). This fault tolerance capacity has been achieved by using interleaving [15]. More information can be seen in [14].

### **FUEC-xAEC**

Next group of ECC used was introduced in [16]. In this case, these ECC are designed to protect 16-bit data words. ECC implemented are:

- FUEC-DAEC. Using 7 parity bits, this ECC can correct single and double adjacent errors, as well as it can detect 3- and 4-bit burst errors.
- FUEC-TAEC. In this case, using 8 code bits, this ECC can correct single and double adjacent errors, as well as 3-bit burst errors. In addition, it can detect 4-bit burst errors.
- FUEC-QUAEC. This ECC can correct single and double adjacent errors, as well as 3- and 4-bit burst errors using 9 redundant bits.

## **3 Results obtained**

### **3.1 Redundancy analysis**

When designing an ECC, one of the most important parameters is the number of extra bits that are added. These bits, called parity bits, code bits, or redundant bits, are generated by the encoder and used by the decoder to detect and/or correct possible errors.

These extra bits must be stored together with each one of the memory words, so it is important to achieve a low number of them. Anyway, a greater error coverage commonly requires a higher number of parity bits. In this work, we are protecting a determined number of variables, so parity bits are not stored in each memory word. As the number of variables augments, much more attention should be put on this parameter. Table 1 summarizes the number of data and code bits for each ECC used in this work.

An aspect to consider is that FUEC-DAEC, FUEC-TAEC and FUEC-QUAEC are designed for 16-bit data words. In the case of hardware implementation, two encoders and two decoders would be used, each encoder/decoder pair processing 16 bits. In the case of this work, we have processed the lower part of the data word first, and then the upper part.

To check the behavior of the different ECC, we have carried out a set of fault injection experiments by instrumentalizing the software. We have implemented a small fault injection routine able to emulate bit-flips in the protected variables. We have injected single and multiple adjacent faults, as well as burst errors. Neither the software size nor the execution time of the fault injection routine have been included in the tables of next subsections.

### **3.2 Software size analysis**

Table 2 shows the software sizes obtained after implementing the different Ultrafast codes for both systems (in parenthesis, the overhead percentage produced with respect to the non-protected system). We can see that software size grows with the error coverage. This is an expected result, as a higher error coverage is accomplished with

more complex formulas during encoding and decoding processes. Specifically, the biggest software size corresponds to SEC-DED-20AEC. In the case of the STM32F429i-DISCOVERY card, this software size is affordable, as it has 64 MB of memory.

In the case of the Arduino UNO, we can observe that the ECC with the highest fault tolerance capacity (SEC-DED-20AEC) cannot be implemented (only 32 KB of memory is available). Also, the SEC-DED-16AEC version can be implemented, but it occupies almost all memory.

We can see also that software size in the Arduino UNO system is lower than in the STM32F429i-DISCOVERY card. This is caused by the specific bit functions of the Arduino programming language [16]. This can be seen more clearly in the overhead percentages. They are smaller for STM32F429i-DISCOVERY than for Arduino UNO. As just commented, these numbers are caused by the specific programming constructions allowed in each system.

**Table 2.** Size of software

System	ECC	Size (KB)
STM32F429i-DISCOVERY [10]	Non-Protected System	21.0
	SEC-DED-8AEC	29.0 (38.10%)
	SEC-DED-12AEC	33.4 (59.05%)
	SEC-DED-16AEC	35.0 (66.67%)
	SEC-DED-20AEC	40.6 (93.33%)
Arduino UNO [11]	Non-Protected System	9.72
	SEC-DED-8AEC	25.12 (158.44%)
	SEC-DED-12AEC	29.26 (201.03%)
	SEC-DED-16AEC	31.32 (222.22%)
	SEC-DED-20AEC	--

### 3.3 Execution time analysis

Table 3 presents the execution time of the Ultrafast ECC implemented (in parenthesis, the overhead percentage produced with respect to the non-protected system). As expected, a greater fault tolerance carries out a greater execution time. As the temperature and humidity sensor used generates new data every 2 seconds, both systems can execute data protection without problems of temporization.

In general, software execution time is lower for the STM32F429i-DISCOVERY card. As this element has a greater clock frequency, software execution is faster. In the case of the Arduino UNO, we can see small differences between the execution times of each code. Bit functions of Arduino environment allows a fast and efficient software execution, causing these small differences. This can be observed in the overhead percentages. We can see that the overhead of the slowest ECC (SEC-DED-16AEC) is lower than 1%.

**Table 3.** Execution time

System	ECC	Execution time (ms)
STM32F429i-DISCOVERY [10]	Non-Protected System	159.2
	SEC-DED-8AEC	175.3 (10.11%)
	SEC-DED-12AEC	189.8 (19.22%)
	SEC-DED-16AEC	202.0 (26.88%)
	SEC-DED-20AEC	233.7 (46.80%)
Arduino UNO [11]	Non-Protected System	532.8
	SEC-DED-8AEC	536.5 (0.55%)
	SEC-DED-12AEC	537.0 (0.79%)
	SEC-DED-16AEC	537.2 (0.83%)
	SEC-DED-20AEC	--

### 3.4 Deeper analysis of Arduino UNO

We have extended our study by implementing more ECC in the Arduino system, as Arduino is extensively used in IoT. As seen before, Arduino has a smaller memory capacity and a lower clock frequency than the STM32F429i-DISCOVERY. Specifically, we have implemented a series of ECC with lower fault tolerance capacities (FUEC-xAEC). Also, we have deactivated the Double Error Detection in the SEC-DED-xAEC codes (resulting in codes called SEC-xAEC). Finally, we have also implemented the Hamming and OLS ECC, well-known ECC used intensively. Results can be seen in Table 4 and Table 5 (in parenthesis, the overhead percentage produced with respect to the non-protected system).

**Table 4.** Size of software (Arduino UNO)

System	ECC	Size (KB)
Arduino UNO	Non-Protected System [11]	9.72
	Hamming SEC	13.56 (39.51%)
	OLS SEC-DEC	17.11 (76.03%)
	SEC-8AEC [11]	16.57 (70.47%)
	SEC-12AEC [11]	18.06 (85.80%)
	SEC-16AEC [11]	19.30 (98.56%)
	SEC-20AEC [11]	22.63 (132.82%)
	FUEC-DAEC [11]	13.20 (35.80%)
	FUC-TAEC [11]	15.11 (55.45%)
	FUEC-QUAEC [11]	16.31 (67.80%)

We can see that all these ECC can be executed in the Arduino UNO system. With respect to the software size (Table 4), Double Error Detection property of SEC-DED-xAEC codes need very complex formulas that provoke big size software. Without this

characteristic, even the ECC with the highest fault tolerance fits in the Arduino UNO memory. About FUEC-xAEC codes, they have a lower size, as their fault tolerance is also lower, so they fit without problems in the Arduino memory. Finally, Hamming code presents a low size, as its fault tolerance is also low, whereas OLS code presents a high overhead. This ECC was designed to get a low delay, so it presents a high redundancy, that provokes this great overhead.

**Table 5.** Execution time (Arduino UNO)

System	ECC	Execution time (ms)
Arduino UNO	Non-Protected System [11]	532.75
	Hamming SEC	533.83 (0.20%)
	OLS SEC-DEC	537.59 (0.91%)
	SEC-8AEC [11]	535.67 (0.55%)
	SEC-12AEC [11]	535.85 (0.58%)
	SEC-16AEC [11]	535.98 (0.61%)
	SEC-20AEC [11]	536.00 (0.61%)
	FUEC-DAEC [11]	534.43 (0.32%)
	FUC-TAEC [11]	534.92 (0.41%)
	FUEC-QUAEC [11]	535.24 (0.47%)

Regarding the execution time (Table 5), differences are minimal. We can see that the overhead introduced by the different ECCs is lower than 0.91%, so all ECC meet the 2 seconds period of the humidity and temperature sensor.

## 4 Conclusions

In this work, we have implemented different ECC with a wide variety of fault tolerance capacities in two typical IoT devices. The idea has been to check the viability of this protection against possible memory faults. Although ECC implemented in this work were originally designed for hardware, we have checked that it is possible to implement them in software. As we are protecting a limited number of variables instead of the complete memory, it is viable using ECC with a great number of redundant bits. Also, we have checked that this protection is effective for single and multiple faults.

This work opens the opportunity to use low-cost IoT devices in systems where reliability is needed. Specially, in the Arduino system, we have seen that the temporal overhead is minimal, an important issue in real time behaviors.

Next step will be testing more complex control systems, including more sensors and actuators, which would lead to an increase in the number of variables to be protected. In addition, other types of ECC, and implementations specifically designed and optimized for software, will be checked. Also, it is necessary a deeper study on the power consumption when ECC are added.

## References

1. "Cisco Global Cloud Index: Forecast and Methodology, 2016–2021", White Paper, 2018.
2. F. Mattern & C. Floerkemeier, (2010). From the Internet of Computers to the Internet of Things. In From active data management to event-based systems and more (pp. 242-259). Springer, Berlin, Heidelberg.
3. K. Cao, Y. Liu, G. Meng and Q. Sun, "An Overview on Edge Computing Research," in IEEE Access, vol. 8, pp. 85714-85728, 2020, doi: 10.1109/ACCESS.2020.2991734.
4. M. Satyanarayanan, "The emergence of edge computing," Computer, vol. 50, no. 1, pp. 30-39, Jan. 2017.
5. W. Shi, J. Cao, Q. Zhang, Y. Li and L. Xu, "Edge Computing: Vision and Challenges," in IEEE Internet of Things Journal, vol. 3, no. 5, pp. 637-646, Oct. 2016, doi: 10.1109/JIOT.2016.2579198.
6. International Roadmap for Devices and Systems. [Online]. <https://irds.ieee.org/editions> (Accessed March 14, 2022).
7. G. Tsiligiannis et. al., "Multiple Cell Upset Classification in Commercial SRAMs", IEEE Transactions on Nuclear Science, vol. 61, no. 4, August 2014.
8. E. Fujiwara, Code Design for Dependable Systems: Theory and Practical Application, Ed. Wiley-Interscience, 2006.
9. <https://www.maximintegrated.com/en/products/microcontrollers/MAX32670.html> (Accessed May 10, 2023).
10. J. Gracia-Morán, P. Martín-Tabares, C. Martínez-Ruiz, L.J. Saiz-Adalid, "Analysis of overheads caused by adding Error Correction Codes in Embedded Systems", Workshop on Innovation on Information and Communication Technologies (WIICT 2022), pp. 8-15, Valencia, Spain, July 2022.
11. J. Gracia-Morán, L.J. Saiz-Adalid, "Análisis del impacto de la inclusión de Códigos Correctores de Errores en un Sistema Empotrado basado en Arduino", VI Jornadas de Computación Empotrada y Reconfigurable (JCER2022), Jornadas SARTECO, 713-718, Alicante, Spain, September 2022.
12. ST-Electronics. "User Manual. Discovery kit with STM32F429ZI MCU". [https://www.st.com/resource/en/user\\_manual/dm00093903-discovery-kit-with-stm32f429zi-mcu-stmicroelectronics.pdf](https://www.st.com/resource/en/user_manual/dm00093903-discovery-kit-with-stm32f429zi-mcu-stmicroelectronics.pdf) (Accessed May 10, 2023).
13. <https://www.elegoo.com/blogs/arduino-projects/elegoo-uno-r3-project-the-most-complete-starter-kit-tutorial> (Accessed May 10, 2023).
14. L.J. Saiz-Adalid, J. Gracia-Morán, D. Gil-Tomás, J.C. Baraza-Calvo and P.J. Gil-Vicente, "Ultrafast Codes for Multiple Adjacent Error Correction and Double Error Detection," IEEE Access, vol. 7, pp. 151131-151143, 2019, doi: 10.1109/ACCESS.2019.2947315.
15. L.J. Saiz-Adalid, P. Gil-Vicente, J. Gracia-Morán, and J.C. Baraza-Calvo, "Using Interleaving to Avoid the Effects of Multiple Adjacent Faults in On-Chip Interconnection Lines", 14th European Workshop on Dependable Computing (EWDC 2013), pp. 198-201, May 2013.
16. J. Gracia-Moran, L.J. Saiz-Adalid, D. Gil-Tomás, and P.J. Gil-Vicente, "Improving Error Correction Codes for Multiple Cell Upsets in Space Applications", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 26(10), pp. 2132-2142, October 2018.
17. A. Sánchez-Macián, P. Reviriego, J. Tabero, A. Regadío, and J.A. Maestro, "SEFI protection for Nanosat 16-bit Chip On-Board Computer Memories", IEEE Transactions on Device and Materials Reliability, DOI 10.1109/TDMR.2017.2750718, 2017.
18. M.Y. Hsiao, D.C. Bossen, R.T. Chien, "Orthogonal latin square codes", IBM J. Res. Develop, vol. 14, 4, pp. 390-394, 1970.
19. S. Liu, L. Xiao, Z. Mao, "Extend orthogonal Latin square codes for 32-bit data protection in memory applications", Microelectronics Reliability, vol. 63, pp. 278-283, 2016.
20. <https://www.arduino.cc/reference/en/> (Accessed May 10, 2023).

21. R.W. Hamming, "Error detecting and error correcting codes", Bell System Technical Journal, vol. 29, pp. 147-160, 1950.

# Improving blockchain-associated energy consumption in wireless sensor networks

Maytham S. Jabor, Aqeel S. Azez, J. C. Campelo, A. Bonastre

ITACA Institute. Universitat Politècnica de València, Camino de Vera s/n , València  
{mayaz, aqaz}@doctor.upv.es; {jcampelo, bonastre}@upv.es

**Abstract.** Wireless Sensor Networks (WSNs) are widely used for collecting, communicating, and sharing information in various applications. Achieving confidentiality and security in these systems remains a major challenge due to their limited resources. Nowadays blockchain (BC) technology is one of the most promising technologies. It provides security, avoids centralization, and a trusted third party. However, applying BCs in WSNs is not an easy task because BCs tend to consume a lot of energy, computation and memory resources. In this work, the additional complication of adding BC in WSNs is compensated by an energy minimization strategy, which is basically based on minimizing the processing overhead of generating the blockchain hash value, and encrypting and compressing the data traveling from the cluster heads to the base station to reduce the total traffic, leading to a reduction in energy per node. A specific hardware circuit is in charge for implementing the compression technique, hash generation and data encryption. Its compression algorithm is based on chaotic theory. Results show that consumption can be reduced up to 63% following this proposal, making BC technology applicable in these environments.

## 1 Introduction

Wireless sensor networks (WSNs) are multi-hop self-organizing network systems that consists of many low-cost wireless sensor nodes (SNs) connected through wireless communications. As a significant part of the Internet of Things (IoT), WSNs play an important role in many applications. In most cases, the SNs are operated by a battery and cannot be recharged. Moreover, the SNs may be positioned in hard-to-reach or inaccessible environments and are anticipated to stay operating for several months or years. Therefore, mechanisms to reduce the energy consumption of these nodes and maximize the network life cycle have lately attracted the attention of many researchers [1].

In addition, most WSNs are defenseless against security threats and pose a number of security challenges [2]. Security in WSNs and IoT is a key problem and not an easy one [3]. Security mechanisms require certain resources for their implementation, such as code space, data memory, and energy to power the sensor. But these resources are very limited in small wireless sensors [4].

One type of security technology is a blockchain (BC), introduced by Nakamoto [5] in 2008. The security of blockchain systems is vital for potential users [6]. BCs are seen as a very robust technology for resolving trusted communications in a decentralized manner. The basic power of BC is decentralized, allowing direct transactions point-to-point. It is possible to use this approach in distributed systems, since trust is not required for nodes to perform transactions, because a third party is no longer needed in a blockchain [7]. The definition of blockchain is a chain of valid transaction blocks, each containing the hash of a previous block in blockchain. Once a transaction is verified, it is broadcasted to the network and added to each blockchain copy. Therefore, a blockchain is fundamentally a decentralized, distributed, shared, and immutable database ledger that stores the record of assets and transactions over a peer-to-peer (P2P) network [8].

As stated before, and corroborated by many researchers [9–10], WSN/IoT nodes have three major limitations: battery capacity, computing hardware and memory. This will create complexity in achieving security features in such systems [11,12].

The main cause of power consumption is the process of transmitting and receiving data; hence, the greater the volume of data, the higher the power consumption, because the transmission and reception time will be longer. In addition, the longer the distance between transmitter and receiver, the higher consumption. Moreover, data processing is also an important factor in energy consumption. Data processing includes the following: i) Block creation, ii) Mining (obtaining a consensus), iii) Hash generation, iv) Creation of encryption keys and v) Encryption process and use of complicated algorithms.

This work contributes to reducing the energy consumption of WSNs while ensuring security through blockchain technology. The solution is based on a dedicated hardware design that works on the basis of chaos theory's principle of the compressive sensing (CS) to generate a hash value, data compression and data encryption. The use of hardware-implemented chaotic operations for hash computation is well suited. It provides the recommended statistical properties of the generated random numbers using deterministic formulas, resulting in reduced hardware complexity while preserving reliability compared to probabilistic methods for hardware random numbers generation. In addition, fixed-point hardware is possible when using deterministic formulas, which means further reduction in hardware complexity [13]. As a result, there is no longer necessary to add an algorithm to encrypt data which consumes a lot of resources in terms of time and energy.

To demonstrate the advantages of our hardware approach, the power consumption and WSN lifetime with respect to a WSN provided with blockchain functionality without our dedicated circuit and a WSN with blockchain implementation and CS techniques implemented in software will be evaluated.

## 2 Proposed System

This section describes the proposed circuit to provide a solution when implementing a blockchain in a WSN. This circuit will be responsible for performing the data compression and security functions (blockchain related functions) to reduce the load on node processors and power consumption. The compression process uses the signal compression technique with chaos to select the samples while considering encryption, while the authentication operations include hash generation.

### 2.1 Circuit design and implementation

The two main function of the dedicated circuit are: i) signal compression and ii) the hardware-implemented algorithms for encryption and hash generation. The following subsections describe their main characteristics.

### 2.2 Signal compression

The first function of the designed circuit is to compress the data based on chaos theory for sampling. Compressive sensing (CS) requires random sampling. Usually, uniform, Gaussian or Bernoulli distributions are used. However, the generation of these random sensing matrices is complicated and requires high memory storage. Chaos is an alternative and less complicated way to generate these sensing matrices, which are generated using a deterministic approach with a few parameters that can control the sending and receiving parties. In addition, the use of chaos introduces a level of security due to the sensitivity of chaos to its parameters. The control parameters consist of three values: length of the chaotic sequence  $L$ ,  $a$ , and the initial value  $X_0$ . They control the determination of the outputs of the chaotic system by certain initial states of these parameters. These chaotic equations are created through a simple chaos system, comprising the logistical map determined by Eq. (1).

$$X_{n+1} = a X_n(1 - X_n) \quad (1)$$

The signal compression algorithm using chaotic logistic map is:

```

Input: Input L,  $\alpha$ , b, d, sos, v[], x[0]
Output: Y (compressed data)
1: for i = 1: L-1
2: x[i]  $\leftarrow \alpha$  x[i-1] *(1-x[i-1])
3: end for
4: for i = 0: L - 1
5: x[i]  $\leftarrow b^*$  (x[i]+d)
6: end for
7: i = 0
8: while (i<sos)
9: for j = 0: L - 1
10: i  $\leftarrow i$  *x[j]
11: if (i< = sos)
12: Y $\leftarrow$  v[i]
13: end if
14: end for
15: end while

```

where  $X_0$  is the initial value of chaotic logistic map, L represents the length of the chaotic sequence,  $\alpha$  is a control parameter defined as a value to determine the appearance and behavior of the logistic map, d and b are constant set to the maximum required sampling interval, SoS represents the length of the original signal, V is the input signal and Y represents the value of the resulting compressed data.

Experimentation with this technique show a compression rate equal to 51.167% with a standard deviation equal to 0.077.

### 2.3 Encryption and hash value generation

Chaotic compressive sensing is designed to achieve simultaneous compression and encryption [14,15]. This encryption has many important features: security analysis showed that this resists brute force attacks, is sensitive to secret keys, robust against statistical attacks, and can use the parameters of the chaotic system as the secret key in the construction of the measurement matrix and also the masking matrix [14]. The computational complexity of these operations is much lower than that of the current main-stream RSA encryption scheme [16]. In addition, the circuit is also responsible for generating the hash value that represents the backbone of blockchain technology and is also considered the main block identifier in the blockchain. This method is proposed because it is characterized by low complexity. The hash is outlined in the following algorithm.

```

Input: L,  $\alpha = 4$ , b, d, sos, v [], x [0],M (message)
Output: f(x) hash value
Hash generation
1: for i = 1: L-1
2:  $x[i] \leftarrow \alpha x[i-1] *(1-x[i-1])$ 
3: end for
4: Divide M into blocks
5: padding message M
6: for j = 0 to n-block
7:  $f(x) = M_i \text{ XOR } x[i]$ 
8: end
9: display f(x)

```

Thus, when cluster-head receives the compressed and encoded data from Sensor nodes, the designed circuit creates a chaotic sequence, computes the hash and then composes the Tx block by encrypting the data with the previous hash.

The above algorithms were coded in Verilog language and synthesized on a FPGA. From the FPGA implementation of the proposed circuit, power consumption results have obtained, as shown in next section. The hash generation and compressed sensing algorithms were also implemented in software to compare the improvements in power consumption with those obtained in hardware.

### 3 Results

In this section, we analyze the performance of our proposal by evaluating its energy efficiency. For this purpose, we will compare our proposal with other versions of a WSN with blockchain but without the dedicated circuit. The simulation and evaluation of the proposed method is performed in MATLAB version 2019b environment using the results obtained from the implementation of the circuit in the FPGA and the power consumption of the software versions.

The power consumption of the implemented dedicated circuit for compression is shown in Table 1.

**Table 1.** Dedicated circuit consumption for CS and Hash

Function	Power consumption / Watt	Signal size
CS	0.082W	752 bits
Hash	0.663 W	752 bits

To compare with software-implemented functions, hash SHA256 generation algorithm was implemented in an ultra-low power microcontroller. A STM32L476 from ST semiconductors was selected as a representative microcontroller of actual embedded

applications. Data from ST<sup>1,2</sup> shows a power consumption equal to 28  $\mu\text{A}/\text{MHz}$  using an external switched-mode power supply (SMPS) and operating at 24 MHz. This is the minimum consumption in an operational stage. With this consumption per MHz, the best results (minimum consumption) can be obtained. Additionally, results were also obtained for a more common case. An experiment was performed with the microcontroller running at 80 MHz (its maximum frequency) and without external switched-mode power supply. This could be representative of many actual designs.

Table 2 presents the power consumption for SHA256 algorithm implemented by software in the STM32L476 microcontroller.

**Table 2.** Software implemented versions. Power consumption

Algorithm	24 MHz with SMPS		80 MHz without SMPS	
	Time	Energy	Time	Energy
SHA (3800 bits of input data)	120 ms	0,0002611J	36 ms	0,00114998J

Our proposal will be compared as follows:

- a) We will obtain the energy consumption and lifetime of a WSN with the Leach algorithm and a traditional software-implemented blockchain (i.e., no data compression or software-obtained hash calculation).
- b) We will compare the above results with a WSN with Leach's algorithm and a software-implemented blockchain (i.e., without the dedicated circuit) but with the chaos compression algorithm. The compression technique (chaos) should be responsible for part of the improvements in network lifetime. Therefore, we decided to measure the advantages and disadvantages (additional consumption) of implementing chaos compression in software.
- c) Finally, we will obtain the results of our proposal: a WSN with a Leach algorithm and hardware implemented support (dedicated circuit) for blockchain operations.

For each of the above scenarios, the network lifetime has been obtained. The network lifetime can be described as the time elapsed from the start of the simulation to the time when the battery power of the last node does not allow its operation (the node dies). In all cases the initial number of nodes is 100.

<sup>1</sup> [https://www.st.com/resource/en/application\\_note/an4978-design-recommendations-for-stm32l4xxx-with-external-smps-for-ultra-low-power-applications-with-high-performance-stmicroelectronics.pdf](https://www.st.com/resource/en/application_note/an4978-design-recommendations-for-stm32l4xxx-with-external-smps-for-ultra-low-power-applications-with-high-performance-stmicroelectronics.pdf)

<sup>2</sup> [https://www.st.com/resource/en/product\\_presentation/stm32l4\\_marketing-pres.pdf](https://www.st.com/resource/en/product_presentation/stm32l4_marketing-pres.pdf)

**Table 3.** Network lifetime

Technique	Number of rounds (microcontroller at 24 MHz)	Number of rounds (microcontroller at 80 MHz)
Reference: standard WSN with Leach protocol (without blockchain)	2576	2576
WSN with Leach protocol and software implemented BC	1184	765
WSN with Leach protocol and software implemented blockchain with software chaos compression	2013	1655
WSN with Leach protocol and hardware implemented blockchain (dedicated circuit)	2199	2199

As can be seen in Table 3, a standard WSN based on one the most used routing/grouping protocols (Leach) but without blockchain capabilities has a network lifetime equal to 2576 rounds: fewer operations, less consumption, and longer lifetime. But obviously, it does not benefit from the desired advantages of blockchain technology for securing the system.

By using a security technology such as blockchain, it will increase energy consumption, which will shorten the lifetime of the network. The blockchain implemented as software without data compression has a network lifetime equal to 1184 or 765 rounds (for 24 MHz and 80 MHz respectively). On the other hand, when our proposed chaos compression was used as software, there was a clear improvement in network lifetime, from 1184 to 2013 and from 765 to 1655 rounds. This improvement increased significantly when our proposal was used as hardware compared to the traditional blockchain or the proposal as software: 2199 rounds. It is close to those of the non-blockchain version. Thus, the use of blockchain becomes a real possibility as the network lifetime is not reduced so drastically.

From an energy consumption point of view, Table 4 shows the benefits of software functions to compress data to support BC operations and also the benefits of a specific hardware support.

**Table 4.** Energy gain (%)

Technique vs technique		Energy gain
WSN with Leach protocol and software implemented blockchain with software chaos compression	WSN with Leach protocol and software implemented BC (24 MHz)	39,80%
WSN with Leach protocol and software implemented blockchain with software chaos compression	WSN with Leach protocol and software implemented BC (80 MHz)	52.01%

WSN with Leach protocol and hardware implemented blockchain (dedicated circuit)	WSN with Leach protocol and software implemented BC (24 MHz)	43,89%
WSN with Leach protocol and hardware implemented blockchain (dedicated circuit)	WSN with Leach protocol and software implemented BC (80 MHz)	63,04%

As shown in table 4 (depending on the operating frequency of the microcontroller), software chaos compression offers an energy gain between 39,8 and 52 %. When using the specific designed hardware, the energy gain is up to 63%.

## 4 Conclusions

Adopting blockchain in a WSN is not easy. Several critical limitations (power, memory size, computational load, and energy consumption) need to be addressed. This proposal is a step forward in addressing these important limitations for the adoption of blockchain technology in WSN. To overcome these problems, a dedicated circuit is designed to perform some blockchain-related functions, such as hash generation and data compression. This work has demonstrated how hardware implementation of these complex software functions significantly contributes to reducing the power consumption of the entire network and prolonging its lifetime.

## References

1. Wang J, Tawose OT, Jiang L, Zhao D. A, "New Data Fusion Algorithm for Wireless Sensor Networks Inspired by Hesitant Fuzzy Entropy". *Sensors* (Basel, Switzerland). 2019 Feb;19(4). DOI: 10.3390/s19040784.
2. T. Kgogo, B. Isong and A. M. Abu-Mahfouz, "Software defined wireless sensor networks security challenges," *IEEE AFRICON*, Cape Town, 2017, pp. 1508-1513, doi: 10.1109/AFRCON.2017.8095705.
3. M. Mamdough, M. A. I. Elrukhsi and A. Khattab, "Securing the Internet of Things and Wireless Sensor Networks via Machine Learning: A Survey," *2018 International Conference on Computer and Applications (ICCA)*, 2018, pp. 215-218, doi: 10.1109/COMAPP.2018.8460440.
4. Abdullah Said Alkalbani, T. Mantoro and A. O. M. Tap, "Comparison between RSA hardware and software implementation for WSNs security schemes," *Proceeding of the 3rd International Conference on Information and Communication Technology for the Moslem World (ICT4M)*, Jakarta, 2010, pp. E84-E89, doi: 10.1109/ICT4M.2010.5971920.
5. Nakamoto, Satoshi. "Bitcoin: A peer-to-peer electronic cash system." *Decentralized business review* (2008): 21260.
6. M. Pilkington, "Blockchain technology: principles and applications," *Research handbook on digital transformations*, p. 225, 2016.

7. Nakamoto, Satoshi. "Bitcoin: A peer-to-peer electronic cash system." *Decentralized business review* (2008): 21260.
8. Z. Zheng, S. Xie, H. Dai, X. Chen and H. Wang, "An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends," *IEEE International Congress on Big Data (BigData Congress)*, Honolulu, HI, USA, 2017, pp. 557-564, doi: 10.1109/BigDataCongress.85.
9. Khan, Minhaj Ahmad, and Khaled Salah. "IoT security: Review, blockchain solutions, and open challenges." *Future generation computer systems* 82 (2018): 395-411.
10. K. Karimi and S. Krit, "Smart home-smartp hone systems: Threats, security requirements and open research challenges," in *20 /9 International Conference of Computer Science and Renewable Energies (ICCSRE)*. IEEE, 2019, pp.1-5.
11. Y Yang, L. Wu, G. Yin, L. Li, and H. Zhao, "A survey on security and privacy issue s in internet-of-things," *IEEE Internet of Things Journal*, 4(5), 2017 pp . 1250- 1258.
12. J. Bugeja , A. Jacobsson , and P. Davidsson , "On privacy and security challenges in smart connected homes," in *2016 European Intelligence and Security Informatics Conference (EISIC)*. IEEE, 2016, pp. 172175.
13. Gousia Thahniyath, M. Jayaprasad, "Secure and load balanced routing model for wireless sensor networks", *Journal of King Saud University - Computer and Information Sciences*,2020.
14. Je Sen Teh, Kaijun Tan, Moatsum Alawida, "A chaos-based keyed hash function based on fixed point representation," *Cluster Computing*, Springer, 2018.
15. Ponuma, R., Amutha, R. Encryption of image data using compressive sensing and chaotic system. *Multimed Tools Appl* 78, 11857–11881,2019, <https://doi.org/10.1007/s11042-018-6745-3>.
16. S. Hsieh, T. Hung, C. Lu, Y. Chen and S. Pei, "A Secure Compressive Sensing-Based Data Gathering System via Cloud Assistance," in *IEEE Access*, vol. 6, pp. 31840-31853, 2018, doi: 10.1109/ACCESS.2018.2844184.
17. Z. Yang, W. Yan and Y. Xiang, "On the Security of Compressed Sensing-Based Signal Cryptosystem," in *IEEE Transactions on Emerging Topics in Computing*,3(3), pp. 363-371, 2015, doi: 10.1109/TETC.2014.2372151.

# Hardware Accelerating a Convolutional Neural Network Using High-Level Synthesis

David de Andrés<sup>1</sup> and Juan Carlos Ruiz<sup>1</sup>

ITACA, Universitat Politècnica de València, Campus de Vera s/n, 46022, Spain

**Abstract.** Nowadays, the deployment of deep learning solutions at the edge in industrial, automotive, or medical environments, to cite a few, requires balancing different conflicting goals, like throughput, power consumption, silicon area, and robustness. Hence, the high throughput per watt of programmable-logic devices makes them suitable candidates to act as hardware accelerators for convolutional neural networks. However, it is a daunting task to model such networks at the register-transfer level, using a hardware description language, to implement them on the target technology efficiently. In recent years, high-level synthesis has provided an additional abstraction level, so hardware designers can focus on describing the behaviour of a system, and automated tools generate the detailed micro-architecture. This approach can reduce the cost of developing custom hardware accelerators for neural networks. Still, several parameters, compilation directives, and code transformations should be adequately applied to optimise the resulting implementation. This practical experience report presents a case study of a C description of a LeNet-5 convolutional neural network architecture and its adaptation to generate an optimised hardware implementation using high-level synthesis.

## 1 Introduction

The increasing computing power of modern computers has enabled the application of deep-learning techniques to multiple fields, like computer vision, speech recognition, drug design, and climate science, among others. In many application domains, like automotive, aerospace, industrial or medical, there is an increasing pressure to move the computation to the edge (closer to the device sensing the environment to reduce the latency) rather than relying on cloud computing to analyse the incoming data to make correct decisions [10].

Custom hardware accelerators are commonly used to face the challenge of obtaining the performance per watt required in these environments while reducing the silicon footprints and meeting the expected robustness. *Graphical Processing Units* (GPUs) are power-hungry devices, and *Neural Processing Units* (NPU), despite their power efficiency, cannot be easily customised. *Field-Programmable Logic Arrays* (FPGAs), on the other hand, are power-efficient, can maximise parallelism and can be reconfigured to implement new requirements, correct bugs, or adapt to the specificities of a given circuit [12].

As hardware complexity keeps growing, *Electronic Design Automation* (EDA) tools enable hardware designers to work at higher abstraction levels which are then translated into a physical layout [6]. *Register-transfer level* (RTL) is one of these levels, enabling designers to specify registers and the operations carried out on these registers, using a *Hardware Description Language* (HDL), without worrying about how they will be later translated into a particular target technology. However, this abstraction level is still too low for specific tasks, like *Convolutional Neural Networks* (CNNs), as custom finite states machines must be tailored to fetch and move data across the circuit to feed the different layers of the network continuously, data structures must be optimised to store intermediate values, and operations must be customised for different data types (integer and single-precision floating point number).

*High-Level Synthesis* (HLS) provides a higher level of abstraction and enables designers to describe the circuit functionality and interconnection protocol using some high-level programming language like ANSI C/C++, SystemC or MATLAB, that EDA tools will translate into the corresponding RTL architecture [7]. Not only the productivity of designers is increased by working at a higher abstraction level, but HLS also automatically exploits existing concurrency, inserts registers to limit critical paths, generates the required control logic, implements external interfaces, and efficiently maps computation and storage onto available elements to balance resource usage and bandwidth. Even though HLS tools automatically decide how to translate the program into an RTL architecture efficiently, they usually require user-provided directives and design constraints to guide and optimise the implementation process for each design. Likewise, there is typically limited support for standard libraries, dynamic memory allocation is not supported, recursion is often limited, and system calls are ignored. Accordingly, for HLS to speed up the implementation of efficient hardware kernels, hardware designers must be aware of available compiler directives, design constraints, and key code transformations.

This practical experience report details the process followed to implement a hardware accelerator for a CNN onto an FPGA, describing the key compiler directives and code transformations used and their effect on the resulting implementation. The proposed case study takes a C description of a LeNet-5 CNN architecture [8], which is implemented on an AMD-Xilinx ZCU104 prototyping board using the AMD-Xilinx Vitis HLS tool [2].

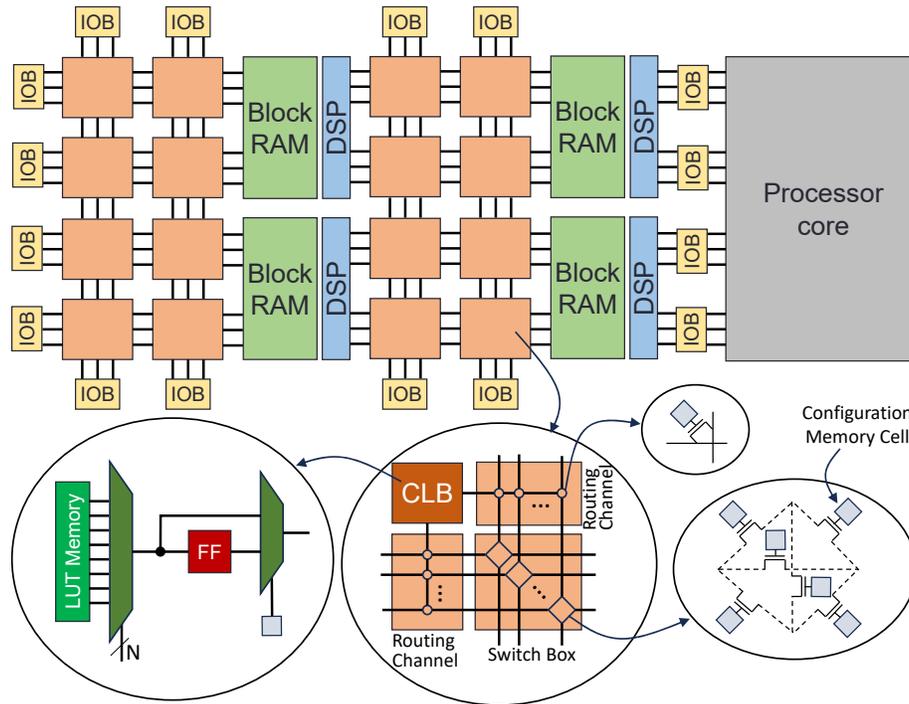
The rest of the paper is structured as follows. Section 2 describes the architecture of FPGAs and the HLS-based design flow. The architecture of LeNet-5 CNN and its C description is detailed in Section 3. The step-by-step transformations on the C code and their impact on the resulting implementation are explained in Section 4. Section 5 verifies the final implementation of the CNN on the prototyping board. Finally, conclusions and future work are discussed in Section 6. selected for the case study

## 2 FPGA Architecture and Design Flow

Implementing optimised hardware accelerators for specific purposes on FPGAs requires a deep understanding of the architecture of these devices and the different processes involved in translating the circuit model from a high-level description into a physical realisation. This section provides the required background on both topics.

### 2.1 FPGA Architecture

As Fig. 1 depicts, FPGAs consist of a two-dimensional array of blocks known as *Configurable Logic Blocks* (CLBs) interconnected by routing channels and switches. This regular fabric is interspersed with several heterogeneous components that provide optimised implementations of commonly used elements [5].



**Fig. 1.** Two-dimensional structure of a generic heterogeneous FPGA

The CLBs comprise an N-input *Look-Up Table* (LUT), which implements combinational logic, and a D-type *Flip-Flop* (FF), which implements sequential logic. LUTs can be seen as small memories that implement any N-variables logic function by using the inputs as address and storing the expected output in the

memory entries. FFs are the primary memory element of the FPGA and store a single bit.

The inputs and outputs of each CLB can be connected to several routing channels. Likewise, *Switch Boxes* enable the interconnection of routing channels following different patterns. All these interconnections are implemented by pass transistors controlled by *configuration memory cell* bits.

Among the heterogeneous elements, memory blocks, multipliers and processor cores are worth mentioning. Memory blocks, known as *Block RAM* (BRAM) by AMD-Xilinx, are configurable random access memory modules that support different memory layouts and interfaces. They typically store large amounts of information internally and transfer information to and from on-chip components. Multipliers, known as *DSP Blocks* (DSP) by AMD-Xilinx, have evolved to efficiently implement multiplication, addition, multiply-accumulate, and word-level logical operations. Finally, some devices may also feature processor cores, usually ARM processors, to deal with sequential operations that cannot be efficiently parallelised and processed by custom hardware. All these elements are also programmable, although they are less flexible than the programmable logic.

Finally, *Input/Output Blocks* (IOBs) connect the routing fabric of the FPGA to the input/output pins of the device. They can also be configured to support different standard interfaces.

All the bits controlling the logic function implemented by LUTs, the internal interconnection of elements within a CLB, the routing of all signals in the circuit, and the configuration of heterogeneous elements and IOBs, should be programmed for the FPGA to provide the desired functionality. This set of bits is known as the *configuration memory* of the device.

## 2.2 HLS-based Design Flow for FPGAs

The purpose of the FPGA-based design flow is to obtain the so-called *bitstream file*. This file is used to program the configuration memory of the FPGA so its internal elements are properly set and interconnected to provide the same behaviour as the modelled system.

The HLS-based design flow mainly changes the first stage of the common FPGA-based design flow [2]:

1. Design entry: The behaviour of the desired circuit is described, as a function, using a high-level programming language, like ANSI C/C++ or SystemC.
2. C simulation: A *testbench*, also written in the same high-level programming language, verifies the functionality of the top-level function.
3. C synthesis: The already verified design is usually modified to include directives that may guide and help the synthesis process to implement specific behaviours or optimisation. Once directives are included (either in the code or by compiler directives) the synthesis process translates the original model into an RTL version described in an HDL (VHDL and/or Verilog).
4. Co-simulation: The resulting RTL model is simulated, using the stimuli provided by the C simulation testbench, and the provided results are checked

against those of the previous C simulation. In this way, it is possible to verify that the intended functionality has not been affected by HLS process.

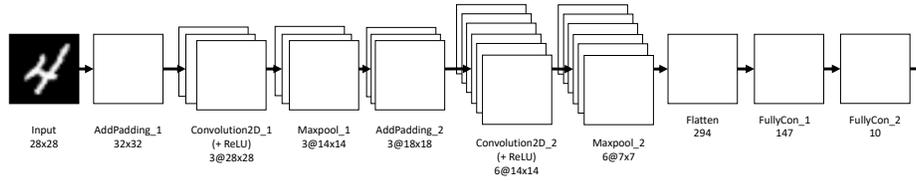
Once the RTL model has been verified, the common FPGA-based design flow is followed [1]:

1. Design entry: The obtained RTL model is used as the entry for the implementation process.
2. Behavioural simulation: A testbench is used to verify the functionality of the RTL model, although this has already been done during the co-simulation process in the HLS flow.
3. Logic synthesis: The model is analysed to determine the basic logic elements required for its implementation and how they must be connected. These elements are mapped to those existing in the underlying technology: LUTs and FFs (later packed into CLBs), BRAMs, and DSPs. The resulting set of interconnected elements is known as a *netlist*.
4. Implementation: The actual components within the device that will implement the netlist are then selected (*placement*), and all the required connections are made (*routing*) through available routing resources so that the timing constraints are met.
5. Timing simulation: The result of the implementation can be simulated, using the same testbench as in step 2 for behavioural simulation, but now including all the timing information provided by the EDA tool (delays of components and interconnections).
6. Bitstream generation: The file that will program the FPGA is generated from the result of the implementation process. It consists of a stream of ‘0’s and ‘1’s that will configure every programmable element of the device to provide the expected functionality.
7. In-device verification: The FPGA is programmed with the generated bitstream to test its behaviour.

### 3 C Description of the LeNet-5 CNN Architecture

LeNet-5 is a CNN architecture defined by LeCun [8] to classify images corresponding to handwritten digits from the MNIST database [9]. The original architecture has been slightly modified, as displayed in Fig. 2, by changing the number of kernels of the different layers and changing the activation from sigmoid to *Rectified Linear Unit* (ReLU), which is more efficient to implement in hardware.

First, a model of this architecture has been defined in PyTorch [11] (see Listing 1.1, to train the CNN and obtain the weights and biases that enable the classification of the ten different kinds of images (numbers from 0 to 9). The MNIST database consists of 10000 images (28x28 greyscale pixels) that can be used for training the model after normalising them by their average (0.1307) and standard deviation (0.3081). Five epochs (training iterations) have been



**Fig. 2.** Adapted architecture of LeNet-5 CNN

executed using a batch of 100 images each to train the model of the network, obtaining an accuracy of 99.9817%

```

1 def createLayers(self):
2     self.conv1 = nn.Conv2d(1, 3, kernel_size=5, stride=1, padding=2)
3     self.relu1 = nn.ReLU()
4     self.max1 = nn.MaxPool2d(kernel_size=2, stride=2)
5     self.conv2 = nn.Conv2d(3, 6, kernel_size=5, stride=1, padding=2)
6     self.relu2 = nn.ReLU()
7     self.max2 = nn.MaxPool2d(kernel_size=2, stride=2)
8     self.fc1 = nn.Linear(7 * 7 * 6, 147)
9     self.fc2 = nn.Linear(147, 10)

```

**Listing 1.1.** Creating the layers of the custom LeNet-5 architecture in PyTorch

Once the weights and biases are obtained from the training process, they are incorporated into a C description of the considered Lenet-5 CNN architecture. As Fig. 2 depicts, the proposal relies on implementing padding, convolution, subsampling (max pooling), and fully connected layers. Sections 3.1 to 3.4 introduce these layers and, Section 3.5 provides the final C-based implementation of Lenet-5.

### 3.1 Padding

Incoming images and features to each convolution layer are padded with two additional rows/columns on each side filled with 0.0. This padding is used to prevent the loss of information that could be located near the edges of images/features when convolved. Listing 1.2 shows the C implementation of the padding layer.

```

1 void addPadding(
2     const float features[FEATURES][HEIGHT][WIDTH],
3     float padded[FEATURES][HEIGHT+2*PADDING][WIDTH+2*PADDING]
4 ) {
5     uint16_t f, h, w;
6     for (f = 0; f < FEATURES; f++)
7         for (h = 0; h < HEIGHT + 2*PADDING; h++)
8             for (w = 0; w < WIDTH + 2*PADDING; w++)
9                 if (w < PADDING || w > WIDTH + PADDING - 1 ||
10                    h < PADDING || h > HEIGHT + PADDING - 1)
11                     padded[f][h][w] = 0.0f;
12                 else
13                     padded[f][h][w] = features[f][h-PADDING][w-PADDING];
14 }

```

**Listing 1.2.** C implementation of the padding layer

### 3.2 Convolution layers

This layer applies a 2D convolution over images with several channels or features. A filter (kernel) slides over the 2D channel/feature, performing an elementwise multiplication and summing up the result into a single output pixel. When the input has several input channels/features, this operation is performed for all the channels/features and results are added together. These layers are in charge of extracting new features from the channels/features.

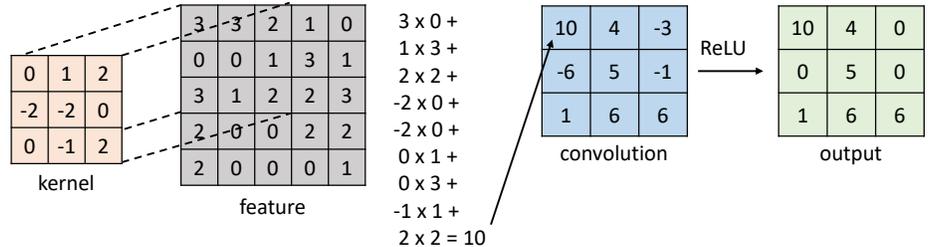
In its simplest case, the output value of the layer with input size  $(C_{in}, H, W)$  and output  $(C_{out}, H_{out}, W_{out})$  is described by Eq. 1, where  $*$  represents the cross-correlation operator,  $C$  denotes the number of channels/features, and  $H$  and  $W$  are their height and width.

$$out(C_{out_j}) = bias(C_{out_j}) + \sum_{k=0}^{C_{in}-1} weight(C_{out_j}, k) * input(k) \quad (1)$$

The cross-correlation operator is defined by Eq. 2, where  $h$  is of size  $(2p + 1)(2p + 1)$ .

$$(g * h)(x, y) = \sum_{m=-p}^p \sum_{n=-p}^p g(x - m, y - n)h(m, n) \quad (2)$$

Fig. 3 depicts an example that convolves a 3x3 kernel with a 5x5 feature and then applies a ReLU activation function to obtain a new feature.



**Fig. 3.** Example of a 2D convolution with ReLU activation

The proposed C implementation of the convolution layer is shown in Listing 1.3. It must be noted that, in this case, the activation function has been integrated within the convolution operation. The selected activation function is ReLU due to its simplicity when implemented on hardware. As can be seen in line 18 of Listing 1.3, it substitutes negative numbers by 0.0 in the result provided by the convolution.

```

1 void convolution2DRelu(
2   const float kernels [KERNELS][FEATURES][KERNEL_HEIGHT][KERNEL_WIDTH],
3   const float bias [KERNELS],
4   const float features [FEATURES][FEATURE_HEIGHT][FEATURE_WIDTH],
5   float output [KERNELS][CONVOLVED_FEATURE_HEIGHT][CONVOLVED_FEATURE_WIDTH]
6 ) {
7   float acc;
8   uint16_t k, fh, fw, kh, kw, f;
9   for (k = 0; k < KERNELS; k++) {
10    for (fh = 0; fh < CONVOLVED_FEATURE_HEIGHT; fh++) {
11     for (fw = 0; fw < CONVOLVED_FEATURE_WIDTH; fw++) {
12      acc = 0.0f;
13      for (kh = 0; kh < KERNEL_HEIGHT; kh++)
14        for (kw = 0; kw < KERNEL_WIDTH; kw++)
15          for (f = 0; f < FEATURES; f++)
16            acc += features[f][fh + kh][fw + kw] * kernels[k][f][kh][kw];
17      acc += bias[k];
18      output[k][fh][fw] = (acc > 0.0f) ? acc : 0.0f;
19    }
20  }
21 }
22 }

```

Listing 1.3. C implementation of the convolution and activation layers

### 3.3 Subsampling layers

Subsampling layers aim to reduce reliance on precise positioning within feature maps, thus preventing weights from being tailored to patterns that are too specific to the training data. Common subsampling layers perform a spatial reduction by substituting a set of pixels with their average (*average pooling*) or maximum (*max pooling*) value, as shown in Fig. 4.

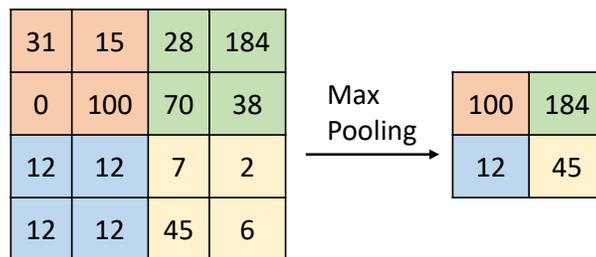


Fig. 4. Example of subsampling a map feature by max pooling

In this case, as shown in Listing 1.4, the maximum value of the set of pixels (STEP x STEP) is used to reduce the size of the feature map.

```

1 void maxPool(
2   const float features[KERNELS][HEIGHT][WIDTH],
3   float output[KERNELS][MAXPOOLHEIGHT][MAXPOOLWIDTH]
4 ) {
5   float maxValue;
6   uint16_t f, hm, wm, hf, wf;
7   for (f = 0; f < KERNELS; f++)
8     for (hm = 0; hm < MAXPOOLHEIGHT; hm++)
9       for (wm = 0; wm < MAXPOOLWIDTH; wm++) {
10        maxValue = 0.0f;
11        for (hf = 0; hf < STEP; hf++)
12          for (wf = 0; wf < STEP; wf++)
13            if (input_features[f][hm * STEP + hf][wm * STEP + wf] > maxValue)
14              maxValue = input_features[f][hm * STEP + hf][wm * STEP + wf];
15        output_features[f][hm][wm] = maxValue;
16      }
17 }

```

**Listing 1.4.** C implementation of the subsampling layers

### 3.4 Fully Connected Layers

These layers are applied after the convolution layers to classify data into defined classes. They apply a linear transformation to incoming data to achieve this goal, as shown in Eq. 3.

$$y = xA^t + b \quad (3)$$

The C implementation of this layer is listed in Listing 1.5.

```

1 void fullyConnected(
2   const float features [FEATURES],
3   const float weights [FCLFEATURES][FEATURES],
4   const float bias[FCLFEATURES],
5   float output[FCLFEATURES]
6 ) {
7   float acc;
8   uint16_t f, nif, nv;
9   for (f = 0; f < FCLFEATURES; f++) {
10    acc = 0.0f;
11    for (nif = 0; nif < FEATURES; nif++)
12      acc += features[nif] * weights[f][nif];
13    output[f] = acc + bias[f];
14  }
15 }

```

**Listing 1.5.** C implementation of the fully connected layers

The output of the last fully connected layer is an array that provides a value for each of the ten categories (from 0 to 9) in which the image can be classified. The image will most likely belong to the category with the maximum value in this array.

### 3.5 LeNet-5

The LeNet-5 CNN architecture can be described in C by calling the previously defined layers following the structure presented in Fig. 2. Listing 1.6 details this description.

```

1 void cnn(
2   const float input_image [FEATURES] [HEIGHT] [WIDTH],
3   float classification [CATEGORIES]
4 ) {
5   float padded_image [FEATURES] [PADDED_HEIGHT] [PADDED_WIDTH];
6   float conv1 [CONV1_KERNELS] [CONV1_FEATURE_HEIGHT] [CONV1_FEATURE_WIDTH];
7   float fconv1_max [CONV1_KERNELS] [CONV1_MAX_HEIGHT] [CONV1_MAX_WIDTH];
8   float conv1_pad [CONV1_KERNELS] [PADDED_MAX1_WIDTH] [PADDED_MAX1_HEIGHT];
9   float conv2 [CONV2_KERNELS] [CONV2_FEATURE_HEIGHT] [CONV2_FEATURE_WIDTH];
10  float conv2_max [CONV2_KERNELS] [CONV2_MAX_WIDTH] [CONV2_MAX_HEIGHT];
11  float flatten [FC1_FEATURES];
12  float fc1 [FC1_FEATURES];
13
14  addPadding_1(input_image, padded_image);
15  convolution2DRelu_1(KERNEL_CONV_1, BIAS_CONV_1, padded_image, conv1);
16  maxPool_1(conv1, first_conv1_max);
17  addPadding_2(conv1_max, conv1_pad);
18  convolution2DRelu_2(KERNEL_CONV_2, BIAS_CONV_2, conv1_pad, conv2);
19  maxPool_2(conv2, conv2_max);
20  flattenLayer(conv2_max, flatten);
21  fullyConnected_1(flatten, WEIGHTS_FC1, BIAS_FC1, fc1);
22  fullyConnected_2(fc1, WEIGHTS_FC2, BIAS_FC2, classification);
23 }

```

**Listing 1.6.** C implementation of the LeNet-5 CNN architecture

The likelihood of belonging to each category can be computed by the *SoftMax* function, as displayed in Eq. 4. However, this function is not considered in this case due to the enormous cost of implementing it in hardware with little benefit. Accordingly, the *cnn* function provides the output of the second fully connected layer. The recipient of this array should determine the index of the maximum value in the array to determine which category the image belongs to with the highest likelihood.

$$Softmax(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}} \quad (4)$$

## 4 Adapting the C Description of LeNet-5 Architecture for HLS

The HLS tool selected to synthesise the C description of the LeNet-5 CNN architecture is AMD-Xilinx Vitis HLS [2]. Even though this tool supports C++, several unsupported constructs must be taken into consideration when describing the C model of the circuit:

- System calls cannot be synthesised and will be ignored.
- To synthesise a hardware implementation, the design must be fully self-contained, specifying all required resources. Accordingly, dynamic memory cannot be used.
- Pointer casting is supported only between native C/C++ types.
- Arrays of pointers to scalars or arrays of scalars are supported, but not arrays of pointers to other pointers.

- Function pointers are not supported.
- Recursive functions cannot be synthesised.
- Specific defined RTL behaviour cannot be expected from an undefined C/C++ behaviour (like non-initialised variables).

Functions and loops are the main focus of most optimisations in the C program. Each function is converted into a hardware component consisting of several small components corresponding to other basic functions. Writing the C program as a set of communicating functions can lead to inferring parallelism while these functions are executed. Since the body of loops is iterated over several times, this can also be exploited to achieve better parallelism. Thus, the loop code may be restructured to prevent data dependencies across iterations from limiting the achievable parallelism.

This section will review the previously presented functions, focusing on the directives and transformations required to implement an efficient hardware accelerator from the HLS perspective.

#### 4.1 Padding Layers

The padding layers do not require any intervention from the designer, as the HLS tool completely pipelines the execution of all the loops.

The C description is automatically pipelined by the HLS tool and achieves an *Iteration interval* (II) of 1. As II denotes the time between consecutive iterations of a loop, a new iteration can start every clock cycle. Its iteration latency is 10, so each iteration will finish after ten cycles. This cannot be helped as data must be read from global memory, which is costly (*gmem\_load\_req* takes eight cycles to complete). Accordingly, the inner loops (lines 7 and 8 from Listing 1.2) have a total latency of 1032 clock cycles (padded images have  $32 \times 32 = 1024$  pixels).

Likewise, the same inner loops are pipelined in the second padding layer with an II of 1 and an iteration latency of 2. Now, as the required values are already in the device’s memory, loading the data needed takes only two clock cycles (*first\_conv\_max\_load(read)*). Thus, as the padded features resulting from the first convolution stage have  $18 \times 18$  pixels, and there are three features, the latency of this second layer is  $18 \times 18 \times 3 = 972$  clock cycles.

#### 4.2 Convolution Layers

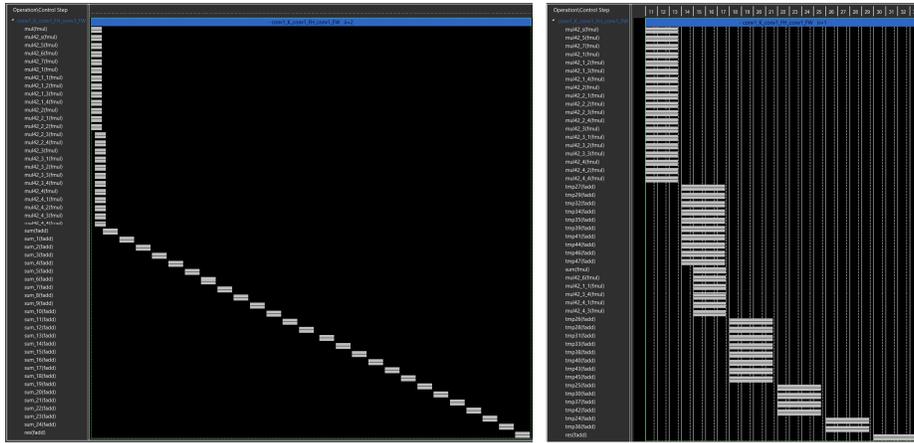
The convolution layers present many problems when synthesising them into an RTL description. These issues slightly differ between the first and second convolutions so they will be presented in two different sections.

**First Convolution** The first attempt at synthesising the first convolution layer reports an II violation when performing the multiply and accumulate operation described in line 16 of Listing 1.3. This operation cannot be wholly pipelined

to start a new execution in the following clock cycle because of a memory dependency with the accumulated value. Accordingly, new multiplications cannot start until the result has been accumulated.

The first step in solving this problem is defining a directive in code that informs the HLS tool that the third loop should only be pipelined with an initiation interval of 2 (a new iteration can start every two clock cycles). The following directive must be set after line 11 in Listing 1.3: `#pragma HLS PIPELINE II=2`.

As the left chronogram in Fig. 5 shows, the first convolution is now synthesisable. It presents an iteration latency of 122 clock cycles, an II of 2, and a trip count of 2352 (3 kernels (5 x 5) applied to padded images (32 x 32 pixels) = 3 x 28 x 28 = 2352 iterations) for a total latency of 4823 clock cycles.



**Fig. 5.** Scheduling of the first convolution: after setting the initiation interval of the third loop to 2 (left), and after generating a tree adder and reducing the initiation interval to 1 (right)

It can be noted that, as new iterations do not start every single cycle, there is no need for performing all the multiplications in parallel. So the HLS saves resources by implementing only half the required DSPs (5x5 multiplications, but only 13 DSPs) and reuses these resources to execute multiplications in subsequent clock cycles.

Likewise, a very long set of operations (sums) run sequentially from clock cycles 14 to 115. They correspond to the accumulation operation in the inner loop and lead to the long iteration latency. Due to saturation and rounding effects in floating point format,  $A=B*C$ ;  $D=E*F$ ;  $O=A*D$ ; may not provide the same result ( $O$ ) as  $A=B*F$ ;  $D=E*C$ ;  $O=A*D$ , so the HLS is unable to reschedule floating point sums to build an adder tree.

This issue can be addressed by i) modifying the code to store all the products in a buffer and manually building an adder tree to compute the sum of

all those products or ii) relying on available directives in the HLS tools. The results obtained by both approaches are very similar, so let us rely on the tool's capabilities.

The compiler flag `unsafe_math_optimizations` must be set to enable rescheduling floating-point operations. After that, the following directive must be inserted into the code, after line 15 of Listing 1.3, to generate an adder tree for this particular operation: `#pragma HLS EXPRESSION_BALANCE`. With this, the iteration latency is reduced from 122 to just 38 clock cycles.

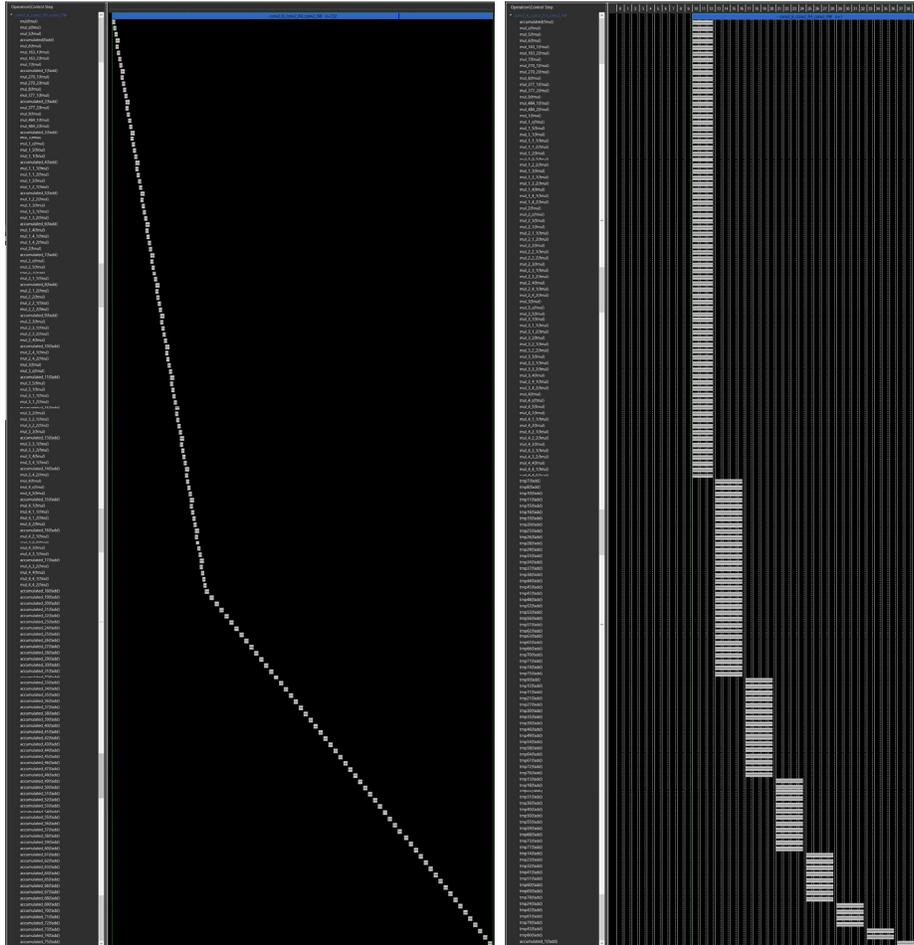
This adder tree has also enabled the previously problematic loop to be completely pipelined. So, by modifying the previously inserted directive in line 11 of Listing 1.3 to `#pragma HLS PIPELINE II=1`, all products can now be computed in parallel, at the cost of increasing the number of required DSPs but starting a new iteration every cycle. However, to do so, the DSPs must be all fed simultaneously with their corresponding inputs. This can only be achieved by reorganising the structure of the input image and weights arrays to retrieve all 25 values simultaneously from memory. Inserting the following directives after line 6 of Listing 1.3 will divide the second and third dimensions of the array of input features into 5 storage elements each, interleaving elements from the original array, thus providing simultaneous read access to all of them: `#pragma HLS ARRAY_PARTITION variable=features type=cyclic factor=5 dim=3` and `#pragma HLS ARRAY_PARTITION variable=features type=cyclic factor=5 dim=2`. Likewise, `#pragma HLS ARRAY_PARTITION variable=weights type=complete dim=4` and `#pragma HLS ARRAY_PARTITION variable=weights type=complete dim=3` are required to get access to all 25 weights of a kernel simultaneously. However, the elements of the array are already in the suitable order, so the directive `type` is changed from *cyclic* to *complete*.

The resulting schedule is displayed in the right chronogram in Fig. 5. Finally, the iteration latency is 36 clock cycles, with `II=1` and a trip count of 2352, for a total latency of 2388 clock cycles, representing a speed up of 100%. However, this is not for free, as the number of DSPs has increased from 12 to 20 (+200%), the number of FFs from 7417 to 7333 (+22%), and the number of LUTs from 10573 to 13354 (+42%).

**Second Convolution** Like in the case of the first convolution, a memory dependency prevents the third loop from starting a new iteration every clock cycle, and an `II` violation of 232 is reported.

After including the `#pragma HLS PIPELINE=232` directive after line 11 in Listing 1.3, the convolution is synthesised but performs poorly. It obtains an iteration latency of 312 clock cycles and `II` of 232 with a trip count of 1176 (6 kernels (5 x 5) applied to padded features (14 x 14 pixels) = 6 x 14 x 14 = 1176 iterations). The inner loops perform 75 multiply and accumulation operations (a 5 x 5 kernel over 3 features), resulting in a long `II`. Thus, the latency of this convolution is 272911 clock cycles.

The left chronogram in Fig.6 shows two problems. On the one hand, the accumulation is performed sequentially using a single adder from clock cycle 76 to 306.



**Fig. 6.** Scheduling of the second convolution: after setting the initiation interval of the third loop to 232 (left), and after generating a tree adder and reducing the initiation interval to 1 (right)

As we already know, this problem can be addressed by enabling the compiler flag `unsafe_math_optimizations` and inserting the following directive after line 15 of Listing 1.3: `#pragma HLS EXPRESSION_BALANCE`. By allowing this directive to generate the adder tree, the iteration latency is 107, and the II has also been reduced to 107, resulting in a total latency of 125834 clock cycles.

On the other hand, no DSP has been assigned to the multiplication operations, which have been implemented using LUTs. This is not a problem by itself, but a new product is performed each cycle (from clock cycles 2 to 75) instead of executing as many operations as possible in parallel. As before, let us reduce II to 1 to compute all the products in parallel (`#pragma HLS PIPELINE II=1`). To do so, it is also necessary to get simultaneous access to all 75 weights and pixels used in the inner loops. Accordingly, the following directives must be set after line 6 of Listing 1.3: `#pragma HLS ARRAY_PARTITION variable=features type=complete dim=1`, `#pragma HLS ARRAY_PARTITION variable=features type=cyclic factor=5 dim=3`, `#pragma HLS ARRAY_PARTITION variable=features type=cyclic factor=5 dim=2`, `#pragma HLS ARRAY_PARTITION variable=kernels type=complete dim=4`, `#pragma HLS ARRAY_PARTITION variable=kernels type=complete dim=3`, and `#pragma HLS ARRAY_PARTITION variable=kernels type=complete dim=2`.

The scheduling for the second convolution operation can be seen on the right chronogram of Fig.6. Now, it achieves an iteration latency of 43, with an II=1, which leads to a total latency of just 1219, representing a speed up of 223%. However, a staggering amount of resources is required to build this component: 223 DSPs (none before), 33131 FFs (+300%), and 41322 LUTs (+900%).

Whether this is admissible depends on the resources available in the selected FPGA and those required to build the rest of the components of the CNN. This is always a tradeoff between performance and resource utilisation, and the ultimate decision depends on the designer. Table 1 compares synthesis results for different II values.

**Table 1.** Latency achieved and resources used to build the second convolution operation with different II in the third loop.

II	Iteration latency	Trip count	Total latency	DSPs	FFs	LUTs
107	107	1176	125834	0	7330	4169
12	42	1176	14141	0	8427	4580
5	53	1176	5927	8	18215	12031
1	43	1176	1219	223	33131	41322

### 4.3 Subsampling Layers

The subsampling layers perform a max pooling operation to reduce the size of incoming features and do not present any major problems when synthesising

them. The loops of both components are automatically pipelined and obtain an II of 1.

The first subsampling component achieves an initiation interval of 6, and an II of 1 with a trip count of 588 (3 features (28 x 28 pixels) with step 2 = 3 x 14 x 14 = 588). Likewise, the second subsampling component achieves an initiation interval of 6, and an II of 1 with a trip count of 294 (6 features (14 x 14 pixels) with step 2 = 6 x 7 x 7 = 294).

#### 4.4 Fully Connected Layers

Fully connected layers follow a pattern very similar, but much simpler, to that of convolution operations, as only two loops are involved because the incoming features are previously flattened.

Thus, the first attempt to synthesise the first fully connected component also fails with an II violation of 3 due to a memory dependency related to the accumulation operation. After inserting the directive `#pragma HLS PIPELINE II=3` after line 9 in Listing 1.5, this operation is synthesised. It achieves an iteration latency of 1185 cycles, with an II of 3 and a trip count of 147 (number of weights), for a total latency of 1622 clock cycles. All 294 multiplications (flattened features = 6 x 7 x 7 = 294) are computed in parallel, but the accumulation is sequentially computed, as before, resulting in the long iteration latency.

To speed up the accumulation process, the adder tree generation is enabled using the compiler flag `unsafe_math_optimizations` and inserting the following directive after line 11 of Listing 1.3: `#pragma HLS EXPRESSION_BALANCE`. It is also necessary to gain access to all 49 pixels and weights as they will be required in the inner loops to execute all products in parallel. The following directives, inserted after line 6 of Listing 1.5 takes care of that: `#pragma HLS ARRAY_PARTITION variable=features type=cyclic factor=49 dim=1`, and `#pragma HLS ARRAY_PARTITION variable=weights type=cyclic factor=49 dim=2`. This reduces the iteration latency to 45 cycles for a total latency of 482 clock cycles, representing a speed up of 3. In addition, it reduces to an 80% the total number of FFs (67277) and LUTs required (37759) to implement this layer. This is achieved by using a total of 365 DSPs and 295 BRAMs.

It is possible to achieve an II of 1 by modifying the related directive. However, to reduce the latency from 482 to 186 clock cycles it requires 1345 DSPs (+268%), 124035 FFs (+84%), and 94125 LUTs (149%). This is roughly the 77% of the DSPs available in the selected target FPGA, so a balance must be found between maximising the performance and using available resources.

The synthesis of the second fully connected layer follows a similar path, with an initial II violation of 4 due to a memory dependency, which is solved by inserting the directive `#pragma HLS PIPELINE II=4` after line 9 in Listing 1.5. In this case, to simultaneously access the required pixels and weights the dividing factor for the input arrays is 37 (147 / 4 = 36.5), resulting in the following directives inserted after line 6 of Listing 1.5: `#pragma HLS ARRAY_PARTITION variable=features type=cyclic factor=37 dim=1`, and `#pragma HLS ARRAY_PARTITION variable=weights type=cyclic factor=37 dim=2`.

This results in an iteration latency of 600 cycles, with  $II = 4$  and a trip count of 10, for a total latency of 635. This latency can be further reduced by activating the compiler flag `unsafe_math_optimizations` and inserting the following directive after line 11 of Listing 1.3: `#pragma HLS EXPRESSION_BALANCE`.

With this, the iteration latency is reduced to 40 cycles for a total latency of 80 clock cycles, which is 8 times faster than with  $II=4$ . In this case, no DSPs or BRAMs are used in the implementation, only 24325 FFs and 6532 LUTs.

Now, the  $II$  can be reduced to 1, which reduces latency from 80 to 47, but at the cost of using 245 DSPs, and increasing the number of FFs and LUTs in +49% and +174%, respectively. Clearly, it is not wise to go *all out* for performance without considering the implications regarding the resources required to implement such architecture.

## 4.5 Synthesised LeNet-5

After properly configuring all the directives for each of the constituting components, it is necessary to define the interfaces to be used for receiving the incoming data (images of 28 x 28 pixels) and sending the generated results (10 values, one for each classification category).

The required directives must be inserted after line 4 of Listing 1.6, one defining an AXI4-Lite interface to control the operation of the component, and two defining a common AXI4 interface to send and receive data: `#pragma HLS INTERFACE mode=s_axilite port=return bundle=control`, `#pragma HLS INTERFACE mode=m_axi port=image depth=1*28*28 bundle=gmem`, and `#pragma HLS INTERFACE mode=m_axi port=classification depth=10 bundle=gmem`.

The resulting latency and estimated resource utilisation, for the whole CNN and each layer in isolation, are listed in Tables 2 and 3. The former table accounts for the results obtained after synthesising the original C description, whereas the latter lists the results obtained after including and configuring all the discussed compiler directives. As a reference for comparison, this implementation takes up to 57% of the BRAMs, 28% of the DSPs, 29% of the FFs, and 50% of the LUTs available and the selected FPGA for experimentation.

The estimated maximum clock frequency for this design is 136.99MHz.

## 5 Results

The prototyping board used for verifying the functionality provided by the final implementation is an AMD-Xilinx ZCU104 kit (see Fig 7). It features an AMD-Xilinx XCZU7EV-2FFVC1156 FPGA with a quad-core ARM Cortex™-A53 applications processor, dual-core Cortex-R5 real-time processor, Mali-400 MP2 graphics processing unit, 4KP60 capable H.264/H.265 video codec, 504K system logic cells, 38Mb of memory, 1728 slices, and 64 user input/output pins.

This section details the results of the CNN implementation for the selected FPGA and its execution on the prototyping board.

**Table 2.** Latency achieved and estimated resources used to build the whole CNN without any optimisations

Layer	II	Total latency	BRAMs	DSPs	FFs	LUTs
First padding	1	1034	0	0	630	566
First convolution	2	4731	0	0	8220	6285
First subsampling	1	594	0	0	625	783
Second padding	1	974	0	0	50	467
Second convolution	3	282240	1	0	158	531
Second Subsampling	1	300	0	0	59	692
First fully connected	4	172887	83	1	499	389
Second fully connected	4	5898	4	1	469	422
Lenet-5	-	468974	155	67	12909	14224

**Table 3.** Latency achieved and estimated resources used to build the whole CNN after configuring compiler directives

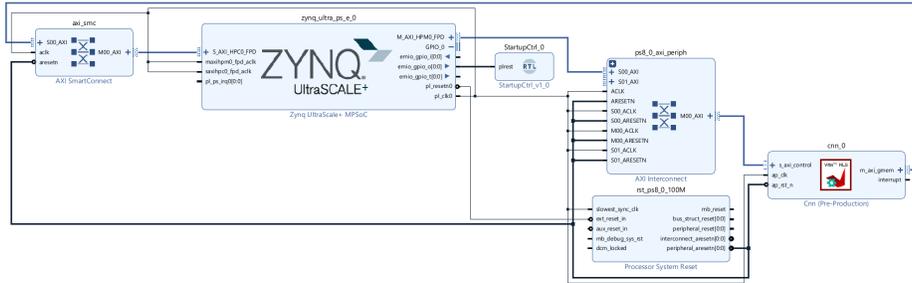
Layer	Total latency		BRAMs	DSPs	FFs	LUTs
	II	(Speed up)				
First padding	1	1034 (x1)	0	0	529	1135
First convolution	1	2388 (x2)	0	0	5063	11254
First subsampling	1	594 (x1)	0	0	387	518
Second padding	1	982 (x1)	0	0	832	2057
Second convolution	1	1219 (x232)	0	6	15160	23920
Second Subsampling	1	300 (x1)	0	0	377	502
First fully connected	3	488 (x354)	295	115	49546	21209
Second fully connected	4	89 (x66)	0	0	24426	6735
Lenet-5	-	7443 (x63)	360	499	134568	115569
(Increased resources)	-	-	(x2.32)	(x3.22)	(x10.42)	(x8.12)

## 5.1 Implementation for the Selected FPGA



**Fig. 7.** AMD-Xilinx Zynq UltraScale+ MPSoC ZCU104 Evaluation Kit

AMD-Xilinx Vivado ML Edition [4] is the tool for realising the RTL description into a physical implementation for the selected device. After creating a new Vivado project for the target prototyping board, the output of the HLS synthesis (the RTL model of the custom LeNet-5 architecture in VHDL and Verilog, packed an Intellectual Property (IP) core) is imported into the project. The *IP Integrator* tool has been used to model the design displayed in Fig. 8.



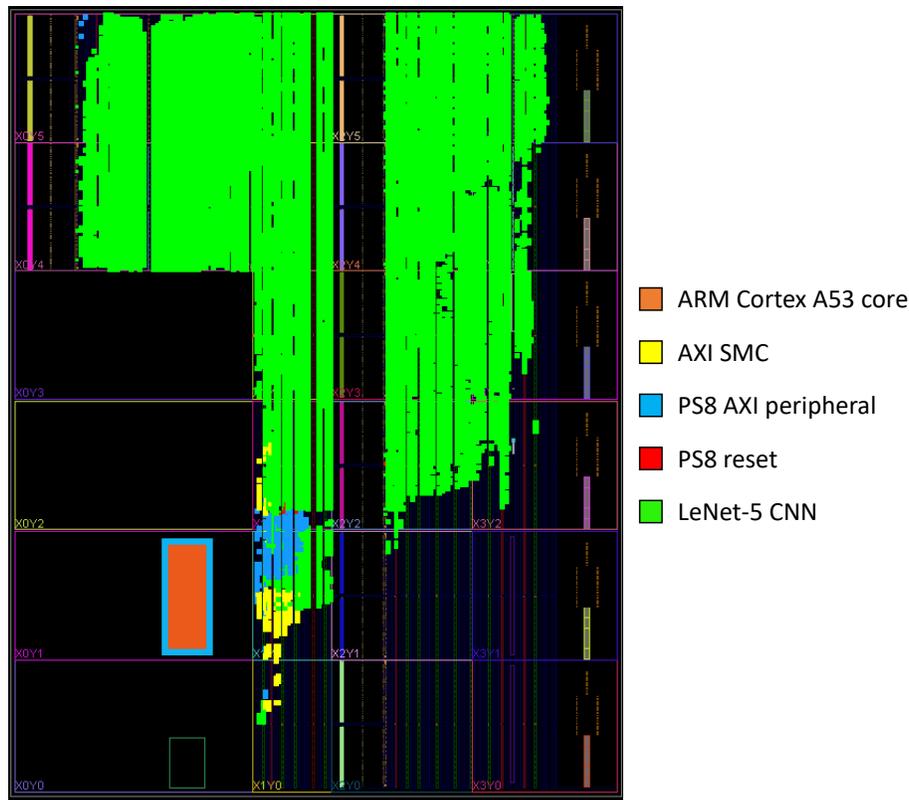
**Fig. 8.** Schematic diagram of the interconnection of the ARM core and the RTL description of the LeNet-5 architecture

This design uses the AXI master interface of the ARM core available within the chip (*ZYNQ UltraScale+ MPSoC*) to connect to the slave AXI4-Lite interface of LeNet-5 (*Cnn*). This will be the control bus of the system where the processor can send commands to the hardware accelerator and monitor its progress. An adapter (*AXI interconnect*) is required to interconnect these slightly different interfaces.

Likewise, the master AXI4 interface of the LeNet-5 is connected to the AXI slave interface of the ARM core through an adapter (*AXI Smart Connect*). This enables LeNet-5 to retrieve and store data to/from the main memory using DMA to obtain the input image and provide the resulting classification array.

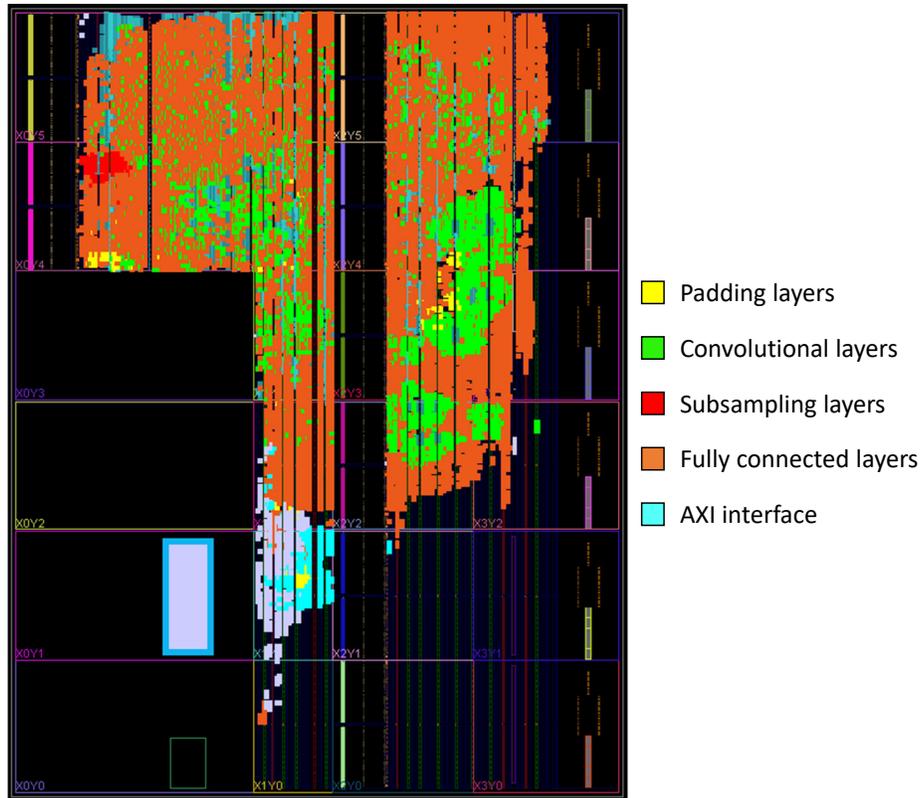
After synthesising and implementing this design for the target board, the actual resource utilisation is listed in Table 4. After optimisations applied during the implementation process, the design takes 50% of the BRAMs, 29% of the DSPs, 23% of the FFs, and 37% of the LUTs available in the device. This is a bit less than what was reported by the HLS process, which must be considered as an upper bound.

Fig. 9 highlights the final physical space taken in the FPGA by the components shown in the schematic diagram of Fig. 8. As expected from the resource consumption report, LeNet-5 takes most of the space, and a limited amount of resources are devoted to its interconnection with the ARM core, which is located on the bottom-left part of the chip



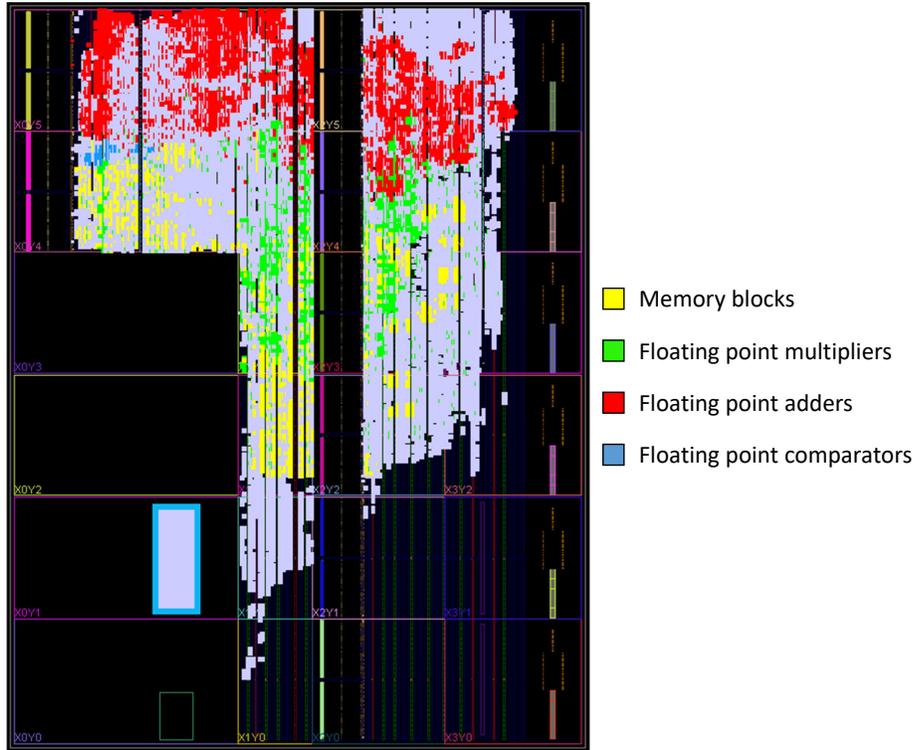
**Fig. 9.** Physical allocation of resources for integrating the ARM core and the CNN implementation within the FPGA

Fig. 10 details the location of the components belonging to each layer of the CNN within the FPGA. The fully connected layers take most of the resources, followed by the convolutional layers. The subsampling layers are located on the left side of the chip, whereas the AXI interface is located on its bottom part, close to the ARM core, to reduce the communication latency.



**Fig. 10.** Physical allocation of resources for each layer of the CNN implementation

Finally, Fig. 11 displays the allocation of hard components (BRAMS and DSPs) for the CNN implementation of the FPGA. As can be seen, memory blocks are distributed across the design to store the results of each layer. Multipliers are also evenly distributed, whereas adders are mostly located on the top of the FPGA because they are heavily used by both fully connected layers to speed up their execution. Finally, comparators are located just on the left part of the chip, as they are only used by the subsampling layers to compare floating point values and determine the maximum value.



**Fig. 11.** Physical allocation of BRAMs and DSPs for the CNN implementation

The timing report provides a *Worst Negative Slack* (WNS) of 0.64 ns for the target clock period of 10 ns, implying a maximum clock frequency of  $1/(10 - WNS) = 106.84MHz$ .

The power consumption report, displayed in Fig. 12, details the power consumption according to the different physical components use for implementing the whole system on the FPGA. The total power consumption estimation amounts to 6.126 W, with 2.741 W consumed by the processing system (ARM Cortex A-53 core) and 3.385 W consumed by the programmable logic (implementation of the CNN and AXI interfaces).

In addition to the bitstream file to program the configurable logic of the FPGA, the hardware configuration of the ARM core must also be exported to generate a board support package containing the required drivers to compile programs for the ARM processor that could interact with the hardware accelerator.

## 5.2 On-Board Verification

This board support package is imported into the AMD-Xilinx Vitis tool [3] to enable embedded software development for the ARM core.

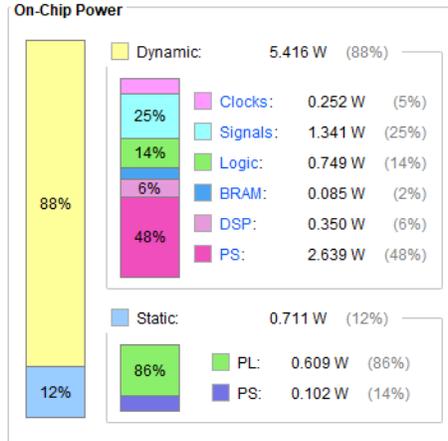


Fig. 12. Power consumption estimation for the implemented CNN

Table 4. Resources used to implement the design in the prototyping board

Name	BRAMs	DSPs	FFs	LUTs
Synthesis				
auto_pc	0	0	481	417
auto_ds	0	0	916	864
zynq_ultra_ps_e	0	0	0	265
axi_smc	0	0	946	128
rst_ps8	0	0	40	19
xbar	0	0	144	198
cnn	156	500	105644	94290
Implementation	156	500	107891	86017

A C program, shown in Listing 1.7, has been developed to store images in main memory, notify the hardware accelerator that can start its execution (so it will directly access main memory to retrieve the image), and monitor the control bus waiting for the hardware accelerator to signal the end of the inference. After that, the classification values will be retrieved from the main memory and compared to determine the maximum value. This will denote the category selected as most probable for the input image.

After compiling this software, the resulting binary file was used to program the prototyping board. The ARM core executes the C program, whereas the bitstream file configures the programmable logic to implement the design CNN and accelerate the inference process (classification of incoming images). The result, retrieved through UART and displayed on the PC screen, shows that all 100 images considered in the workload for experimentation are successfully classified.

The original C-description of LeNet-5 has also been compiled for both the ARM Cortex A-53 available within the FPGA, as a representative of an embedded processor for executing software at the edge, and on an Intel i7 CPU, as a representative of a more powerful but also power-hungry computer. The time required to process 100 images and the estimated power consumption for the three considered platforms are listed in Table 5.

**Table 5.** Execution time to execute 100 images and estimated power consumption

Platform	Clock frequency (MHz)	Execution time (ms)	Power consumption (W)
ZCU104	106.837	62.99	6.126
Intel i7-4790	3800	93.75	47.5
ARM Cortex-A53	1200	1271.74	2.74

The most power-efficient solution uses the ARM processor but is also the slowest. The implemented hardware accelerator doubles the estimated power consumption of the ARM core but achieves a speed-up factor on x20 even though its clock frequency is 12 times lower. It even outperforms the Intel-based solution, with a speed-up factor of x1.5 and nearly 8 times less power consumption. It achieves this with a clock frequency 38 times lower.

As can be seen, the hardware accelerator is the best platform from a throughput-per-watt perspective, and it can be a suitable solution to integrate be integrated at edge solutions.

```

1 static const float IMAGES[100][1][28][28] = {...};
2 static const int prediction[100] = {...};
3 XCnn instancePtr;
4
5 void initMem(int num_image) {
6     u32* base = (u32*) XPAR.PSU_DDR_0.S.AXLBASEADDR;
7     u32 offset = 0x0;
8     u32 data;
9     for (int i = 0; i < 28; i++) {
10        for (int j = 0; j < 28; j++) {
11            data = *(unsigned int*) &INPUT_IMAGES[num_image][0][i][j];
12            *(base + offset) = data;
13            offset++;
14        }
15    }
16 }
17
18 int checkResult(int num_image) {
19     u32* base = (u32*) XPAR.PSU_DDR_0.S.AXLBASEADDR;
20     u32 offset = 0x0;
21     u32 read;
22     float data;
23
24     float max;
25     int index;
26
27     read = *(base + offset);
28     offset++;
29     max = *(float*) &read;
30     index = 0;
31     for (int i = 1; i < 10; i++) {
32         read = *(base + offset);
33         data = *(float*) &read;
34         if (data > max) {
35             max = data;
36             index = i;
37         }
38         offset++;
39     }
40     if (index == res_cnn_prediction[num_image]) return 1;
41     else return 0;
42 }
43
44 int main() {
45     int results[100];
46     init_platform();
47     XCnn_Initialize(&instancePtr, XPAR.CPU_ID);
48     XCnn_DisableAutoRestart(&instancePtr);
49     for (int num_image = 0; num_image < 100; num_image++) {
50         while (!XCnn_IsIdle(&instancePtr));
51         XCnn_Set_input_image(&instancePtr, 0);
52         while (!XCnn_IsIdle(&instancePtr));
53         initMem(num_image);
54         XCnn_Start(&instancePtr);
55         while (!XCnn_IsDone(&instancePtr));
56         results[num_image] = checkResult(num_image);
57     }
58     printf("CHECKED_IMAGES:\n");
59     for (int i = 0; i < 100; i++) {
60         printf("%d", results[i]);
61     }
62     printf("\n");
63     cleanup_platform();
64     return 0;
65 }

```

**Listing 1.7.** Embedded software for controlling the operation of the hardware accelerator

## 6 Conclusions

Programmable logic devices promise to accelerate the execution of computationally heavy algorithms with a more efficient performance per watt than general-purpose processors and GPUs. Compared to tensor processing units, they are highly reconfigurable and can adapt to correct bugs, increase functionality, or react to changing environments. Their main problem is that the desired circuit must be described at a register-transfer level, which could be daunting for demanding applications.

HLS has evolved over the years and is now a more mature technology that can ease the development effort of hardware accelerators. This practical experience report has shown how a hardware accelerator for a CNN can be easily described in C and, after including specific compiler directives to guide the HLS process, an RTL description is obtained.

It is necessary to understand the underlying architecture of the selected device, and how the HLS process works, to apply transformations in the original C code to enable loop unrolling and pipelining to achieve the highest throughput. However, increasing performance is usually at the cost of using more resources. This could require exploring different configurations for the selected compiler directives to find a trade-off between the resources available in the target device and the desired performance.

Several courses of action are opened for further research.

It is possible to achieve a higher acceleration by using *stream* interfaces in the functions to represent unbounded, continuously updating data sets. By combining it with the `#pragma HLS DATAFLOW` directive, to enable task pipelining, it could be possible to overlap the execution of functions (layers), increasing the overall throughput of the design when processing a continuous flow of images.

The current version of the CNN uses floating-point numbers and operations, which take lots of time and resources to process. A quantised version of the same CNN which uses integer numbers of 8 bits for data and operations can improve the obtained throughput and reduce resource consumption. However, this may impact the accuracy of the provided classification.

Finally, design space exploration techniques can be deployed to determine the most suitable configuration of the compiler directives to find a particular trade-off between throughput and resource consumption.

## Acknowledgements

This work has been supported by grant PID2020-120271RB-I00 funded by MCIN/AEI/ 10.13039/501100011033.

## References

1. UltraFast Design Methodology Guide for FPGAs and SoCs. Tech. Rep. UG949, AMD-Xilinx (2023), <https://docs.xilinx.com/r/en-US/ug949-vivado-design-methodology>

2. Vitis High-Level Synthesis User Guide. Tech. Rep. UG1399, AMD-Xilinx (2023), <https://docs.xilinx.com/r/en-US/ug1399-vitis-hls/Using-Vitis-HLS>
3. Vitis Unified Software Platform Documentation: Embedded Software Development. Tech. Rep. UG1400, AMD-Xilinx (2023), <https://docs.xilinx.com/r/en-US/ug1400-vitis-embedded/Getting-Started-with-Vitis>
4. Vivado Design Suite User Guide: Synthesis. Tech. Rep. UG901, AMD-Xilinx (2023), <https://docs.xilinx.com/r/en-US/ug901-vivado-synthesis/Vivado-Synthesis>
5. Chang, M.L.: Device Architecture. In: Hauck, S., DeHon, A. (eds.) Reconfigurable Computing: The Theory and Practice of FPGA-Based Computation, pp. 3–27. Morgan Kaufmann Publishers Inc., San Francisco, USA (2007)
6. Chen, W.K. (ed.): The VLSI Handbook. CRC Press (08 2022)
7. Kastner, R., Matai, J., Neuendorffer, S.: Parallel Programming for FPGAs. ArXiv e-prints (May 2018)
8. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. *Proceedings of the IEEE* 86(11), 2278–2324 (1998)
9. LeCun, Y., Cortes, C., Burges, C.: Mnist handwritten digit database. ATT Labs. <http://yann.lecun.com/exdb/mnist> 2 (2010)
10. Motahhir, S.: Smart Embedded Systems and Applications. River Publishers (08 2022)
11. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R. (eds.) *Advances in Neural Information Processing Systems*. vol. 32. Curran Associates, Inc. (2019), [https://proceedings.neurips.cc/paper\\_files/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf)
12. Shawahna, A., Sait, S.M., El-Maleh, A.: FPGA-Based Accelerators of Deep Learning Networks for Learning and Classification: A Review. *IEEE Access* 7, 7823–7859 (2019)



The ITACA-WIICT is a meeting forum for scientifics, technicians and other professionals who are dedicated to Information and communication technologies study and research. Its fundamental scope is to promote the contact among scientific and professionals, improving the cooperation as well as the technological transfer among professionals.