UNIVERSITAT POLITÈCNICA DE VALÈNCIA

School of Industrial Engineering

Calculation of the decarbonisation potential of the installation of green roofs in the L'Illa Perduda neighbourhood in València

End of Degree Project

Bachelor's Degree in Industrial Engineering

AUTHOR: Zayas Orihuela, Max

Tutor: Gómez Navarro, Tomás

Cotutor: Bastida Molina, Paula

ACADEMIC YEAR: 2022/2023

Abstract

Cities play a pivotal role in tackling climate change. They are accountable for more than 70% of world CO_2 emissions and consume around 65% of the world's energy. If the world is to address climate change, urban climate neutrality must be a priority. To do so, nature-based solutions can help improve a city's resilience, stabilise temperatures and capture CO_2, both mitigating climate change effects and adapting for them. This study aims to develop a methodology to select valid rooftops for green roof (GR) installation in the city of Valencia and calculate the available area and possible benefits using geographic information systems (GIS) and artificial vision (AV) algorithms. This methodology has been successfully developed and tested using the L'Illa Perduda neighbourhood of Valencia as a case study.

First, cadastral data of the city's buildings' ground plan and its different parts is combined with height measurements from a LiDAR covering to select the rooftops that can be greened. To do so, a $100\,\mathrm{m}^2$ threshold in area and a 10° maximum slope are applied using GIS software. Second, using the selected buildings' ground plan, the total area is calculated, and AV algorithms are applied to high-quality aerial imagery of the neighbourhood to better estimate the area. Third, the potential energy savings and carbon sequestration (both direct through photosynthesis and indirect through non-emitted carbon due to the energy consumption reduction) are calculated. Lastly, an economic analysis is performed to check the viability of the GR installation.

Following the application of this methodology in the L'Illa Perduda neighbourhood of Valencia, about 50% of the buildings are selected as suitable for greening. GR installation on the selected buildings would save 1% of the neighbourhood energy, and sequestrate $354.2\,\mathrm{tCO_2\,yr^{-1}}$ to $484.1\,\mathrm{tCO_2\,yr^{-1}}$. The project of GR installation is not financially viable, since the costs outrun the financial benefits, but could be economically viable if the social benefits are considered.

Keywords: climate change mitigation; climate neutrality; greenhouse gas sequestration; urban decarbonisation; natural-based solutions (NBS); green roofs (GR).

Resumen

Las ciudades deben ser el motor de la transición hacia la mitigación del cambio climático. Más del 70% de las emisiones mundiales de CO_2 provienen de ellas y consumen el 65% de la energía mundial. La neutralidad climática urbana debe ser una prioridad para minimizar las consecuencias del cambio climático. Las soluciones basadas en la naturaleza (SBN) mejoran la resiliencia de las ciudades, estabilizan las temperaturas y capturan CO_2, mitigando los efectos del cambio climático y adaptándose a ellos. Este estudio busca desarrollar una metodología para seleccionar edificios admisibles para la instalación de techos verdes (TV) en Valencia y calcular la superficie disponible y los posibles beneficios asociados a la installación utilizando sistemas de información geográfica (SIG) y algoritmos de visión aritifical (VA). Esta metodología ha sido satisfactoriamente desarrollada y probada utilizando el barrio de L'Illa Perduda, en Valenia, como caso práctico.

Primero, la información catastral pública de los edificios de la ciudad y sus diferentes partes, junto con los datos LiDAR de alturas de la ciudad, fue utilizada para seleccionar los edificios en los que instalar un TV. Para ello, se exige a los tejados una mínima superficie de $100\,\mathrm{m}^2$ y una máxima pendiente de 10° mediante un programa SIG. Después, a partir el plano catastral de los edificios se calcula el área total disponible y mediante el procesado con algoritmos de VA de imágenes aéreas de alta calidad se mejora la estimación. En tercer lugar se calculan los potenciales de ahorro energético y de carbono secuestrado, tanto directa (a través de la fotosíntesis) como indirectamente (por el carbono no emitido a raíz de la energía ahorrada). Por último, se realiza un análisis económico para comprobar la viabiliad de la instalación de TTVV.

Tras la aplicación de la metodología al barrio de L'Illa Perduda de Valencia, aproximadamente el 50% de los edificios son seleccionados como válidos para la instalación de un TV. La instalación de TTVV en los tejados seleccionados implicaría un ahorro del 1% de la energía del barrio y un secuestro de carbono de $354.2\,\mathrm{tCO_2\,yr^{-1}}$ a $484.1\,\mathrm{tCO_2\,yr^{-1}}$. La instalación de GRs no es financieramente viable, ya que los costes superan con creces los ahorros directos, pero puede ser económicamente viable si se consideran los beneficios sociales.

Palabras Clave: mitigación del cambio climático; secuestro de gases de efecto invernadero; descarbonización urbana; soluciones basadas en la naturaleza (SBN); techos verdes (TV).

Resum

Les ciutats han de ser el motor de la transició cap a la mitigació del canvi climàtic. Més del 70% de les emissions mundials de CO_2 provenen d'elles, i consumeixen el 65% de l'energia mundial. La neutralitat climàtica urbana ha de ser una prioritat per a minimitzar les conseqÃ¼ències del canvi climàtic. Les solucions basades en la naturalesa (SBN) milloren la resiliència de les ciutats, estabilitzen les temperatures i capturen CO_2, mitigant els efectes del canvi climàtic i adaptant-se a ells. Aquest estudi busca desenvolupar una metodologia per a seleccionar sostres admissibles per a la instalÂ·lació de sostres verds (SV) a València i calcular la superfície disponible i els possibles beneficis associats utilitzant sistemes d'informació geogràfica (SIG) i algoritmes de visió artificial (VA). Aquesta metodologia ha sigut satisfactòriament desenvolupada i provada utilitzant el barri de L'Illa Perduda, en València, com a cas pràctic.

Primer, la informació cadastral pública dels edificis de la ciutat i les seues diferents parts, juntament amb les dades LiDAR d'altures de la ciutat, va ser utilitzada per a seleccionar els edificis en els quals instalÂ·lar un SV. Per a això, s'exigeix a les teulades una mínima superfície de $100\,\mathrm{m}^2$ i un màxim pendent de 10° mitjanÃ§ant un programa SIG. Després, a partir el pla cadastral dels edificis es calcula el llaure total disponible i mitjanÃ§ant el processament mitjanÃ§ant algorismes VA d'imatges aèries d'alta qualitat es millora l'estimació. En tercer lloc es calculen els potencials d'estalvi energètic i de carboni segrestat, tant directa (a través de la fotosíntesi) com indirectament (pel carboni no emés arran de l'energia estalviada). Finalment , es realitza una anàlisi econòmica per a comprovar la viabilitat de la instalÂ·lació de SSVV.

Després de l'aplicació de la metodologia al barri de L'Illa Perduda de València, aproximadament el 50% dels edificis són seleccionats com a vàlids per a la instalÂ·lació d'un SV. La instalÂ·lació de SVs en les teulades seleccionades implicaria un estalvi de l'1% de l'energia del barri i un segrest de carboni de $354.2\,\mathrm{tCO_2\,yr^{-1}}$ a $484.1\,\mathrm{tCO_2\,yr^{-1}}$. La instalÂ·lació de SSVV no és financerament viable, ja que els costos superen amb escreix els estalvis directes, però pot ser econòmicament viable si es consideren els beneficis socials.

Paraules clau: mitigació del canvi climàtic; segrest de gasos d'efecte d'hivernacle; descarbonització urbana; solucions basades en la naturalesa (SBN); sostres verds (ST).

Contents

List of Figures

List of Tables

List of abbreviations

NSB Natural Based Solution
GR Green Roof
GIS Geographic Information System
AV Artificial Vision
UHI Urban Heat Island
UPV Universitat Politècnica de València
EGR Extensive Green Roof
IGR Intensive Green Roof
SIGR Semi-intensive Green Roof
LCA Life Cycle Analysis
OBIS Object-Based Image Segmentation
PBIS Pixel-Based Image Segmentation
LiDAR Light Detection and Ranging

Part I

Report

Chapter 1

Introduction

1.1 Background

The Paris Agreement was signed during the 2015 United Nations Climate Change Conference. The 194 signatory countries agreed to cooperate towards limiting the temperature increase to 1.5 °C above pre-industrial levels. In December 2019, after the European Commission's communication, the European Council endorsed the goal of achieving climate neutrality by 2050, starting the *European Green Deal Timeline - European Green Deal and fit for 55* 2023. Climate neutrality means reducing greenhouse gas emissions and compensating for the emissions that cannot be diminished, reaching a global net-zero emissions balance (*European Green Deal* 2023). In September 2020, at the European Research & Innovation Days, five EU Missions were proposed. In April 2021, they were launched under the research and innovation programme Horizon Europe (*EU Missions in Horizon Europe* 2023; *Horizon Europe* 2023).

The *EU Mission: Climate-Neutral and Smart Cities* aims to "deliver 100 climate-neutral and smart cities by 2030" and "ensure that these cities act as experimentation and innovation hubs to enable all European cities to follow suit by 2050 ", and Valencia is one of them. Cities are one of the main actors in global climate neutrality. They are accountable for more than 70% of world CO_2 emissions and consume around 65% of the world's energy (*EU Mission: Climate-Neutral and Smart Cities* 2023). If the world is to address climate change, urban climate neutrality must be a priority for climate change mitigation.

Urban climate neutrality is a complex goal, that must be tackled from different perspectives if we are to have any chance of succeeding. The energy networks that supply the cities will be rethought, transitioning to renewable energies and distributed production, and energy consumption shall be reduced by improving the energetic efficiency of products, buildings, infrastructure and human behaviours. The means of transportation will shift towards shared transportation, avoiding fossil fuels. Consumption cycles shall be revisited, waste management improved, and consumer goods made out of recyclable and recycled materials. Cities shall incorporate carbon sinks within their urban environment, be it green areas or higher technological solutions. It will take time, even though there is not much, but it can be done, and lots of work is been put into it.

1.2 Scope and motivation

As it has been stated, Valencia is one of the accepted cities to follow through the European Mission and strive to search for innovative ideas and technological solutions to achieve urban carbon neutrality. The Universitat Politècnica de Valéncia (UPV) has followed suit, aspiring to be the first climate-neutral university in Spain. Both the city and the UPV have joined forces to plan the transition to urban carbon neutrality, as was put forward in the 2021 conference "*La UPV responde a la misión: València Ciutat Neutra*" (*The City Council and the Polytechnic University of Valencia join forces in order to promote the València 2030 Neutral City plan* 2023), and confirmed by the 2022 conference "*La alianza UPV-Ciudad para la Misión Climática València 2030*" (*Conference "The UPV-City alliance for the Climate Mission 2030"* 2022).

This project is one more step toward urban decarbonisation in Valencia, hoping to provide an estimate of the feasibility and the benefits of the green roof (GR) retrofit of the city. Green roofs are included in what is called nature-based solutions (NBS), urban adaptation and mitigation solutions that are based on and employ nature. They are non-intensive interventions and infrastructure in the urban environment that include vegetation and that use mechanisms that occur naturally outside of urban spaces. NBS serve a triple purpose: economic, social, and environmental, and they improve cities' resilience. They help with climate change mitigation and adaptation, create social spaces, and protect cities' biodiversity. They are an instinctive approach to reach urban neutrality that has gained a lot of interest in the past years.

GRs are an interesting NBS, since they connect nature with buildings, improving their thermal properties, capturing carbon, reducing urban temperatures and embellishing the city. This study addresses one of the first solutions that come up when searching for options to achieve carbon neutrality and aims to estimate the benefits that the city could enjoy out of greening its roofs.

1.3 Objectives

The objectives of this project are:

- Develop a methodology for gathering rooftop characteristics and identifying the suitable rooftops for green roof retrofit in the city of València.
- Calculate the benefits that could yield the greening of the city's roofs, from carbon sequestration to energy savings.
- Study the economic viability of such installation.

Chapter 2

Literature review

2.1 Urban climate neutrality

Cities comprise a complex ecosystem in which people and the built environment interact and grow together. Buildings and transport add up to about half the European cities' emissions (Moran et al. 2022). To tackle this issues, the main strategies to achieve climate neutrality fall into two categories: those which aim to reduce greenhouse gas emissions and those which focus on carbon removal from the atmosphere (also called negative emissions) (Chen et al. 2022).

For **carbon emission reduction,** the implementation of *low-carbon policies* can be critical for diminishing emissions, as a study from Jiangsu University and Rana Yassir Hussain shows substantial decreases in several Chinese provinces (Wang et al. 2023), and can be an effective mean to cut off transport-derived emissions. The transition to *clean energy* and energy efficiency—always along with low-carbon policies to prevent further emission increase (Turiel 2020)—is key in urban and global climate neutrality, since GHG emission from electricity generation affects nearly all human activities, making up for scope 2 of GHG emissions as proposed in the Greenhouse Gas Protocol (Ranganathan et al. 2004)[1]. *Buildings* in cities are a major source of direct (construction, maintenance, reform and demolition) and indirect (mainly heating and cooling) emissions. Buildings can reduce their carbon footprint by developing resilient designs, less dependant on electricity (passive solutions), by introducing control systems to improve their energy use efficiency, and by rethinking construction materials, like reintroducing lumber and investing in better building envelopes (Chen et al. 2022).

Atmospheric carbon sequestration is implemented by negative emissions techniques, with *nature-based solutions* as a promising base for urban climate neutrality. NBS directly capture CO_2 through photosynthesis without relying on high technological innovations and are easy to incorporate into the urban environment. A new study showed that NBS in 54 European cities could reduce anthropogenic carbon emissions by an average of 17.4%, and maximum theoretical implementation could reduce urban carbon emissions by up to 25% (Pan et al. 2023).

---

[1] To help delineate direct and indirect emissions, the Greenhouse Gas Protocol defines three scopes for emission accounting and reporting purposes. Scope 1 considers the direct emissions, produced by sources owned by the company. Scope 2 accounts for the electricity-related indirect emissions. Scope 3 gathers all the rest of the indirect emissions.

Nature-based solutions

The European Commission defines NBS as "Solutions that are inspired and supported by nature, which are cost-effective, simultaneously provide environmental, social and economic benefits and help build resilience. Such solutions bring more, and more diverse, nature and natural features and processes into cities, landscapes and seascapes, through locally adapted, resource-efficient and systemic interventions." (*Nature-based solutions* 2023). NBSs are aligned with EU policy, and NBS innovation and research are being backed by the Horizon Europe programme (*Nature-based solutions research policy* 2023).

NBSs are becoming increasingly attractive solutions as knowledge increases and technologies become more cost-effective. Under the scope of Basque Country's project KLIMATEK, the Department of Environment, Territorial Planning and Housing developed a methodological guide to implement, identify and map NBS in the region (Departamento de Medio Ambiente, Planificación Territorial y Vivienda del País Vasco 2016). In this guide, a thorough collection of different solutions is included and each technique is associated with the climatic dangers (floods, sea level rise, droughts, temperature increase and wildfires) it helps to fight and with its benefits from environmental (hydrologic cycle regulation, water quality, soil quality and stabilisation, air quality, acoustic comfort, biodiversity, carbon sequestration), social (health and quality of life, environmental education, social justice), and economic (energy savings, local employment, land value) perspectives.

A more strict definition of NBS was given in a recent article from a collaboration of European researchers (Castellar et al. 2021). They classified the different technological solutions into 32 types, using a 3-level hierarchical structure, separating between NBS units ($\mathrm{NBS_u}$), concrete technologies or green urban spaces, and interventions ($\mathrm{NBS_i}$), the act of intervening in already existing ecosystems or other NBS. $\mathrm{NBS_u}$ are further divided into spatial units ($\mathrm{NBS_{su}}$), as elements of urban green infrastructure per se, and technological units ($\mathrm{NBS_{tu}}$), which are meant to meet certain technological requirements and provide a specific service. Technological units are finally classified as vertical, such as green wall systems or green facades, and horizontal, such as any type of GR.

Zero energy buildings – Passive solutions

In the European Union, buildings account for 40% of the total energy consumption and 36% of energy-related greenhouse gas emissions, with 80% of the energy used in buildings corresponding to heating, cooling and domestic hot water. Moreover, almost 75% of the European building stock is not energy efficient (*Energy performance of buildings directive* 2023). Building energy efficiency is a crucial solution to achieve climate neutrality. Zero energy buildings (ZEB) are highly energy-efficient buildings that, when equipped with a renewable source of energy, become neutral in carbon emissions. The Directive on Energy Performance of Buildings set all new buildings after 2020 to be nearly zero-energy buildings in the EU and addressed member states to establish a long-term renovation strategy to transform their national stock of buildings into a high energy efficient and decarbonised building stock (European Parliment and Council of European Union 2018).

ZEB can be achieved by implementing passive energy-saving technologies, utilising energy-efficient service systems, and installing renewable energy generation techniques (Cao, Dai and J. Liu 2016). **Passive energy saving technologies** are solutions that reduce the energy needs of a building, mainly HVAC, without energy consumption. They mainly target the building envelope, by improving the building insulation, modifying the external surface reflectance, enhancing solar heat capture, modifying the shadowing on the building surface or accentuating internal convection on the building envelope (Sadineni, Madala and Boehm 2011). Passive solutions depend heavily on local climate characteristics, therefore bioclimatic architecture strategies must be considered when implementing these technologies (Manzano-Agugliaro et al. 2015).

Passive building design focuses on walls, fenestration and roofs. **Walls** are a predominant part of a building envelope, especially when a high-rise building is considered. Between the various passive wall technologies, solar walls enhance the heat capture from solar radiation using glazing to take advantage of the greenhouse effect in colder climates, ventilated walls can favour cooling energy savings of up to 40% in the summer (Ciampi, Leccese and Tuoni 2003), and introducing phase change materials into the walls can serve as latent heat storage, decreasing maximum room temperature by 4.2 °C (Kuznik and Virgone 2009). **Roofs** constitute the other major part of a building envelope. They receive direct solar radiation, especially during the summer months in southern Europe, and are most important for building thermal insulation in single-storey buildings. Cool roofs have their external surface treated to boost their reflectivity or albedo and enhance their infrared emittance, achieving 5-40% cooling load savings (Akbari, Levinson and Rainer 2005). GRs are building roofs partially or fully covered with a layer of vegetation. GRs enhance roof insulation, have a high albedo (about 0.7-0.85), favour the building heat loss in hot and dry climates through evapotranspiration, and capture atmospheric CO_2 through photosynthesis.

2.2 Green roofs

GR are NBS technological horizontal units (NBS_{thu}), following J.A.C. Castellar et al. classification (Castellar et al. 2021). They are building roofs covered with a layer of vegetation, artificial ecosystems that serve as passive solutions for thermal comfort in buildings and energy efficiency, as well as modest carbon sinks. The installation of GR (and, in general, of NBS) in the urban environment constitutes a climate change adaptation and mitigation strategy for cities. GRs are strongly coupled with climatic challenges .as water management from pluvial and fluvial floods, temperature increase and heatwaves, and weakly coupled with droughts and other extreme climatic phenomena (Departamento de Medio Ambiente, Planificación Territorial y Vivienda del País Vasco 2016; Mihalakakou et al. 2023). GR **benefits** can be categorised as:

(a) *Energy benefits.* Several studies have assessed the potential of GR to reduce energy consumption in buildings (Borràs et al. 2022; Jaffal, Ouldboukhitine and Belarbi 2012; Peñalvo-López et al. 2020; Saiz et al. 2006; Heidarinejad and Esmaili 2015; Foustalieraki et al. 2017; Castleton et al. 2010). GRs directly increase roof insulation by increasing its total U-value, dampening heat transfer. GRs also can favour heat loss even during hot seasons by evapotranspiration when the soil is wet. GRs must be studied depending on the climate (mean temperature and annual rainfall) since their energy benefits can be similar to those of a highly insulated roof (Fantozzi et al. 2021), but cooling through evapotranspiration can help decrease indoor temperatures and not only insulate during summer (Borràs et al. 2022). GR installation can be especially beneficial to poorly insulated buildings, raising their energy efficiency and also providing multiple co-benefits (Castleton et al. 2010).

(b) *Air quality.* All NBS, as long as they contain the green factor (i.e. presence of vegetation) (Castellar et al. 2021), can contribute to carbon sequestration and general air pollutant removal. There is evidence to CO_2 removal by GR (Shafique, Xue and Luo 2020; Agra et al. 2017; Md. Yacob, Kasmin and Hashim 2021; Ondoño, Martínez-Sánchez and Moreno 2016; Yang, Yu and P. Gong 2008), even though the literature is not extensive. Plant composition and soil materials under different climates can act also as emission sources, so they must be chosen carefully (Velasco et al. 2016). The different pathways GR reduce air pollution are reducing energy consumption, and thus reducing the GHG emitted through non-clean energy sources, fighting urban sprawl, promoting pro-environment behaviour, microclimate regulation, and direct carbon sequestration (Pan et al. 2023).

(c) *Water management.* GR provide a regulating effect on runoff water volume by storing water, delaying the hydrological response. GR also improve rainfall runoff water quality by filtering some of the pollutants and acting as sinks for nitrogen, preventing nonpoint source pollution (W. Liu et al. 2019; Y. Gong et al. 2020; AndrÃ©s-DomÃ©nech et al. 2018).

(d) *Ecosystem and biodiversity.* GRs serve as urban ecosystems, increasing the biodiversity of the cities and improving their habitat connectivity (Oberndorfer et al. 2007; Benedito DurÃ et al. 2023; Donati et al. 2022).

(e) *Temperature control.* Urban heat island (UHI) effect can be countered by urban renaturing. Green infrastructure has a temperature-controlling effect in the cities (Massaro et al. 2023) and GR can help to reduce the UHI effect (Rosenzweig, Gaffin and Parshall 2006), but the effect of the installation of GR in high rise buildings is almost negligible. Buildings with a height-to-street width ratio of 1 or less should be focused on tackling the UHI effect (Santamouris 2014).

(f) *Social, aesthetic and psychological benefits.* GR can support psychological benefits by providing a refuge inside the city. They embellish the city, soften the urban noise, can be perceived as places of emotional respite, and could even improve work performance as their view reduces perceived effort (Williams et al. 2019).

Typical GR comprise (1) a waterproof and a root barrier membrane, (2) a draining layer, (3) a filter membrane, (4) a growing medium (soil), and (5) the vegetation layer (Shafique, Kim and Rafiq 2018). There are three main GR **typologies**, based mainly on the vegetation type and the thickness of the soil layer (Abass et al. 2020; Shafique, Kim and Rafiq 2018; *Normas tecnológicas de Jardinería y Paisajismo - 11C - Ajardinamientos especiales: Cubiertas verdes* 2012):

(a) *Extensive green roofs.* Extensive green roofs (EGR) are light-weight, low-maintenance roofs, characterised by a soil thickness of 7 -10 cm or even up to 15 cm. They sometimes don't need a separate drainage layer and use natural and adaptative vegetation, mainly characterised by moss, succulents (like *Sedum*) or short herbaceous plants. Natural vegetation dynamics help ERG be more resilient and low-maintenance once the vegetation has stabilised, but in hot dry climates, they might need some irrigation during the hottest months. EGR entail a weight load of 60 -150 $\text{kg}\,\text{m}^{-2}$.

(b) *Intensive green roofs.* Intensive green roofs (IGR) have vegetation similar to ground-level green infrastructure, they are often called roof gardens. They need a thick soil layer, over 30 cm, usually above 60 cm and a separate drainage layer. The vegetation can be diverse, even including shrubs and trees, and require abundant irrigation. They need high maintenance and weigh about 200 -600 $\text{kg}\,\text{m}^{-2}$, or even more.

(c) *Seni-intensive green roofs.* Semi-intensive green roofs (SIGR) are a middle ground between EGR and IGR. Their soil layer ranges between 10 to 30 cm, sometimes even more, and their vegetation is a mix of ornamental, meadow species, herbaceous plants, grass and shrubs. They need a separate drainage layer, moderate irrigation, and require regular maintenance. They weight from 100 to 300 $\text{kg}\,\text{m}^{-2}$.

GR installation impose some **technical requirements** on the roofs structural properties (*Normas tecnológicas de Jardinería y Paisajismo - 11C - Ajardinamientos especiales: Cubiertas verdes* 2012; Office of the Chief Building Official 2010; *Green Roof Guidelines - Guidelines for the Planning, Construction and Maintenance of Green Roofs* 2018; Silva, Flores-Colen and Antunes 2017), mainly:

(a) *Structural capacity.* GRs constitute a non-negligible weight load onto the roof-bearing structure. They entail a permanent load—its base weight—and a variable load associated with water absorption. EGR's load is way inferior to those of intensive and semi-intensive roofing, and could probably be installed onto modern buildings' rooftops with none (Saiz et al. 2006) or limited modifications to the current structure. Intensive and semi-intensive roof installation could be more complicated in already-existing buildings. In any case, a thorough structural study should be conducted.

(b) *Roof pitch.* GR are more easily installed in flat or low-pitched roofs. A minimum slope of 2% should be granted to avoid water stagnation, but it is a common requirement in building construction. In Spain a minimum slope of 1% is already required for rooftops (*Código Técnico Estructural (CTE) - Documento básico HS - Salubridad* 2022). Extensive roofing can be installed in pitched roofs, but anti-shear measures should be applied in rooftops with a slope greater than 10° and slopes greater than 20° are usually not recommended (Office of the Chief Building Official 2010), even though GR can be installed with slopes about 30°. Slopes greater than 45° can be greened using techniques closer to those of green walls, and won't be considered as green roofs. Intensive and semi-intensive greening reduces the maximum slope to about 10°, even though some guides allow the slope to reach 20° for semi-intensive roofing (*Normas tecnológicas de Jardinería y Paisajismo - 11C - Ajardinamientos especiales: Cubiertas verdes* 2012).

Life-cycle analysis (LCA) of EGR compared to common flat roofs reduces between 1.0 and 5.3% the environmental impacts in all the categories, most of the reductions associated with the use phase, because of the energy consumption reduction (Saiz et al. 2006)[2]. Other LCA study has shown that urban GR installation needs between 10 and 30 years to balance the pollution produced in its installation, mainly due to emissions from polymer production for drainage and filter layers. The analysis shows that using recycled polymer fabric yields the shortest balance time (around 10 years), about 2.3 times less than with 40% recycled polymer (the most common) and that extensive roofing has a shorter balance period than intensive roofing (Bianchini and Hewage 2012).

2.3 Studies on green roof installation potential

Even though the field is still in development, there are already some studies that develop and apply methodologies to characterise a city's potential for GR installation and to provide tools for urban planning.

- A study from the Disaster Management Institute of Istanbul (Ekmekcioğlu 2023) developed a methodology based on multi-tier decision analysis to **identify the most appropriate GR type for urbanised cities**, and successfully applied it to Istanbul. A list of 16 factors—financial (4), technical (4), knowledge-related (4), and environmental (4)—was used to address the election. The multi-criteria decision analysis revealed that financial and environmental factors were determinants, followed by technical factors, rendering knowledge-related factors negligible. The most important factors were 'Initial an operational cost' (financial), 'Stormwater retention' (environmental), 'Roof lifespan' (financial), and 'Air and water quality improvement and noise reduction' (environmental), followed by 'Structural load (and risk of leakage)' (technical). They conclude that extensive roofing performance ($\mathrm{CC_e} = 0.7442$) within the analysis is higher than that of the intensive option ($\mathrm{CC_i} = 0.2558$) for Istambul.
- Another study from the Civil Engineering Research and Innovation for Sustainability unit of Lisbon (CERIS) (Silva, Flores-Colen and Antunes 2017) proposed a qualitative approach, identifying 19 indexes to **rank existing built areas for GR retrofit**. From this list, a small selection was utilised as the most simple and representative criteria for GR potential estimation: two indexes to characterise the building capacity to incorporate GRs (year of construction and roof slope) and three indexes to evaluate the needs of the area for additional greening (site coverage, green surface area, and urban trees). A 5 step methodology is then proposed, based on these five indexes to determine the city's areas that should be considered first for GR installation. The results of the analysis in Lisbon select 52% of the total city area as suitable for greening and successfully ranked neighbourhoods within the selected area.

---

[2] This result must be clarified, as the electric mix has changed over the last years in Spain, and thus the reduction in environmental impacts associated to energy savings would probably be less if the study was repeated at this moment. Also, the article explains that the HVAC load is further reduced at peak electric demand, when it is likely that additional electricity is supplied, usually by polluting energy sources such as natural gas or coal, so the LCA could be underestimated.

- Some studies address the **potential GR retrofit of a city** (Santos, Tenedório and Gonçalves 2016; Velázquez et al. 2019; Shao et al. 2021). They all calculate the suitable rooftop area for GR installation inside a city or some selection of the city's districts.
    - The **Lisbon analysis** (Santos, Tenedório and Gonçalves 2016) first calculates current vegetation at ground level by calculating the Normalised Vegetation Difference Index (using the vegetation absorption of red light and reflection of infrared light) and later evaluates the potential for roof greening, based on Light Detection and Ranging (LiDAR) to obtain a 3D model of the city and very high-resolution images as well as municipal cartography. The suitable roofs are selected by precluding red-tiled roofs (using block-type feature identification in VHR images) and by establishing a minimum area of $100\,\mathrm{m}^2$, two slope maximums, at 11° and 20°, and three scenarios based on solar incidence, considering roofs that enjoy more than 3 h of sunlight per day, roofs that do not make the requirement and all roofs independently of sunlight incidence.
    - The **analysis performed in Madrid** (Velázquez et al. 2019) calculates the greening potential emphasising the city's green connectivity, using LiDAR data. The study first selects the most suitable neighbourhoods for GR installation by considering four criteria: pollution, existing green areas, traffic and population density and second it chooses the roofs within these neighbourhoods that meet the requirements of (1) height between 4 m and 25 m (to ensure connectivity with urban trees) (2) height standard deviation less than 5 m (from LiDAR data) to ensure roof flatness, and (3) a Probability of Connectivity index greater than 90% when considering roof greening and urban trees.
    - The **study concerning Central Luohe** (Shao et al. 2021) uses ultra-high resolution images taken from an unmanned aerial vehicle to assess GR development potential. Building use and flat or pitched roof typologies are manually established and the available rooftop area is calculated on each category. Based on Luohe's urban law, load-bearing building capacity is assumed for extensive roofing, only flat roofs are considered, and, in principle, only non-residential buildings are taken into account. The results indicate a 4.7% potential increase in green areas in the city, so the study suggests starting greening non-residential buildings (mainly schools, hospitals, offices and hotels) and then considering residential building greening.

  None of the aforementioned studies considers the decarbonisation potential nor the potential energetic savings such an intervention could yield.

- A last **study performed in Amsterdam** (Slootweg et al. 2023) calculates the potential for EGR installation as well as the potential for photovoltaic panels (PV) installation, and for a combination of both technologies (EGR-PV). They only consider slope thresholds for EGR installation (less than 10° for suitable and less than 45° moderate potential) and slope plus roof orientation for PV installation. Their results show that in Amsterdam 42% of the roof are suitable for a combination of both technologies, and only in 3.2% of building roofs one of both solutions should be chosen (competition).

Chapter 3

Methodology

3.1 Scope and data gathering

The present study aims to calculate the decarbonisation potential of GR installation in Valencia, using the L'Illa Perduda neighbourhood as a case study. To do so, the following steps will be performed:

1. *Data gathering.* The analysis will be based on cadastral information of the city of Valencia, a LiDAR covering of the city map and aerial photographs covering the neighbourhood.
2. *Evaluation of the available area.* The calculation of the available rooftop area will be assessed in two parallel ways: using cadastral-based information of the buildings' plans, processed with QGIS, a free and open-access geographic information system; and using artificial vision (AV) processing of the aerial images with MATLAB.
3. *Potential calculation.* The decarbonisation potential assessment will be based on the carbon sequestration potential of GRs and the non-released carbon associated with energy savings, both backed by experimental data.

Figure 3.1 shows the general flow of the methodology.

This study is the first step in evaluating the city's potential to implement green roofing technologies, and as such, it will be a simple estimate of the actual prospects. As it is based on minimal information from the neighbourhood buildings, it will not be exact at a local level, but it is expected to be satisfactorily accurate when the global results are considered. An error of about 20% would be expected, given the estimated nature of the analysis [CHANGE, see section, take off number]. This project also aims to open up the research to AV for rooftop detection in Valencia and could serve as a probe for these methods.

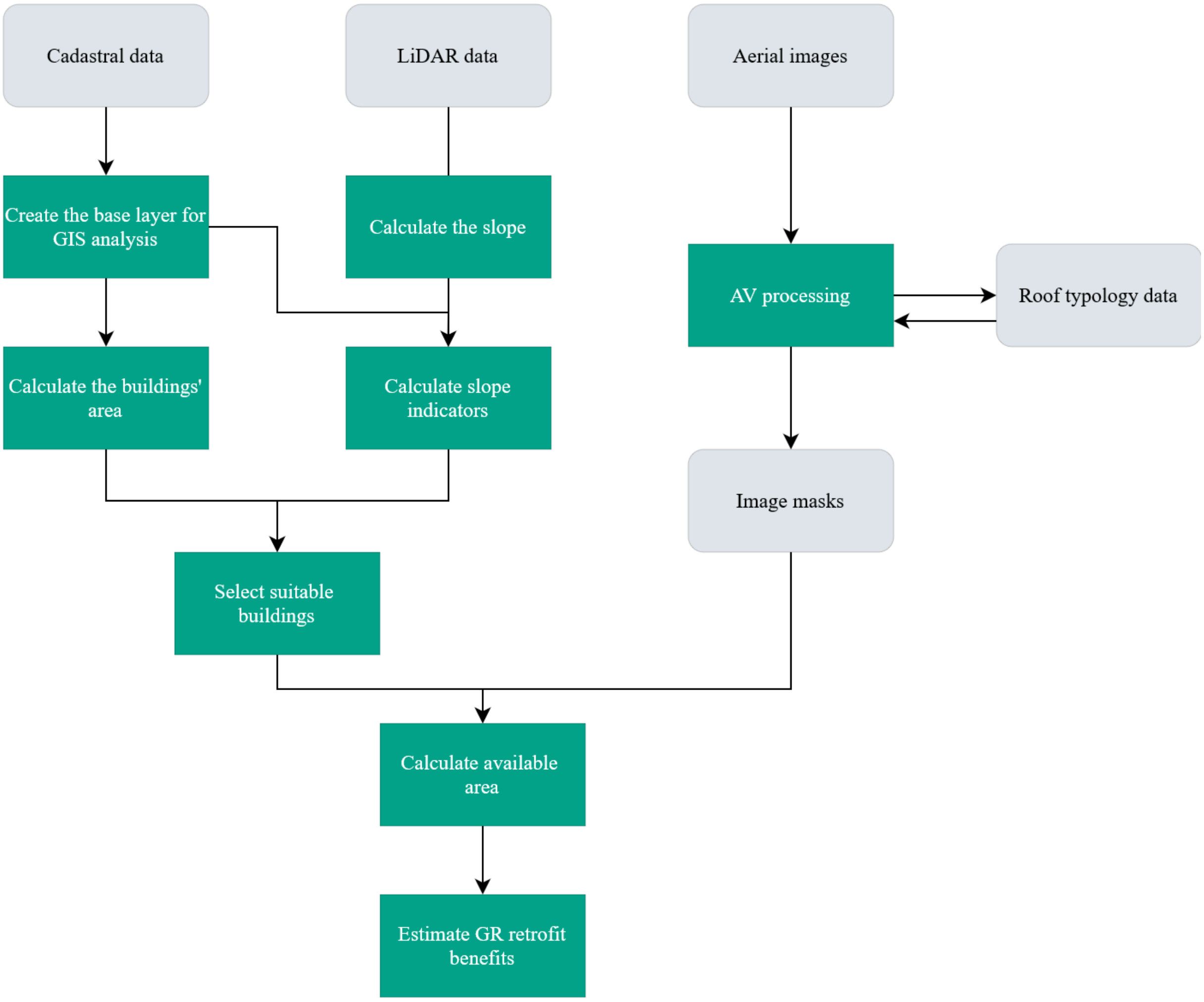


Figure 3.1: Flow diagram for image processing methodology.

The raw data used for the available area computation is:

- *Cadastral information of the city of Valencia.* Two *geoJSON* (an open standard format for representing geometrical geographical features and their associated properties) files were downloaded from the building data from the Spanish repository of cadastral cartography from the European INSPIRE initiative, through the 'Spanish Inspire Catastral Downloader' complement of QGIS (*Servicios INSPIRE de Cartografía Catastral – Edificios* 2023):
  - `A.ES.SDGC.BU.46900.building.geojson`
  - `A.ES.SDGC.BU.46900.buildingpart.geojson`
- *Neighbourhood delimitation of the city of Valencia.* A *geoJSON* file downloaded from the Open Data portal of Valencia's council (*Portal de datos abiertos de Valencia* 2023): `barris-barrios.geojson`.

- *LiDAR 3D map.* Two *LAZ* (a compressed version of the LAS format, an open standard format for handling LiDAR point cloud data) files were downloaded from the Spanish National Center of Geographic Information (CNIG) download centre (*Modelos Digitales de Elevaciones – Mapa LIDAR 2ª cobertura*) to cover the whole neighbourhood (*Centro de Descargas* 2023):
    - `PNOA_2015_VAL_728-4372_ORT-CLA-RGB.laz`
    - `PNOA_2015_VAL_728-4374_ORT-CLA-RGB.laz`
- *Aerial images.* Aerial images of the neighbourhood were taken using Google Earth Pro (LLC 2023), selected so that they cover every building in *L'Illa Perduda*. Thirteen images were needed to patch the entire area.

3.2 Available area calculation

The area calculation was performed in two parallel ways, first using QGIS 3.16.16 (Open Source Geospatial Foundation 2020) and second by AV algorithms implemented on MATLAB (MathWorks 2022). The AV processing is chosen as a more powerful way of estimating available areas and detecting obstacles for GR installation and different roofing solutions. Thus, the parallel evaluation serves a dual purpose: it yields an estimation of the error of the broader calculation associated with obstacles and roofing solutions and at the same time it checks the correctness of the AV-based calculations.

Cadastral-based calculation using QGIS

The QGIS analysis is based on the cadastral information, the neighbourhood delimitation and the LiDAR data. It is performed by modifying several layers of information until only the desired buildings are selected, and then the total potential area is calculated. The steps are the following:

1. *Create the base layer of the neighbourhood's buildings.* The base polygonal layer for area calculations is created by merging both geoJSON layers (`building` and `buildingpart`) and cropping them to the neighbourhood area. The layer `building` contains the current use type for the building (`Building_currentUse` property), while the layer `buildingpart` contains a more detailed plan of the different parts within a building and information concerning the number of floors above (`numberOfFloorsAboveGround` property) and below (`numberOfFloorsBelowGround` property) the ground level. Once the combined layer is made, it is cropped down by the shape of the desired neighbourhood, using the delimitation data from the `barris-barrios` file.
2. *Calculate the building area.* The ground area of each entity from the base building layer is then calculated based on each feature's geometry.
3. *Exclude non-valid buildings.* Once the building layer is set and the surface values are calculated, the building selection has to be filtered to exclude non-suitable roofs.

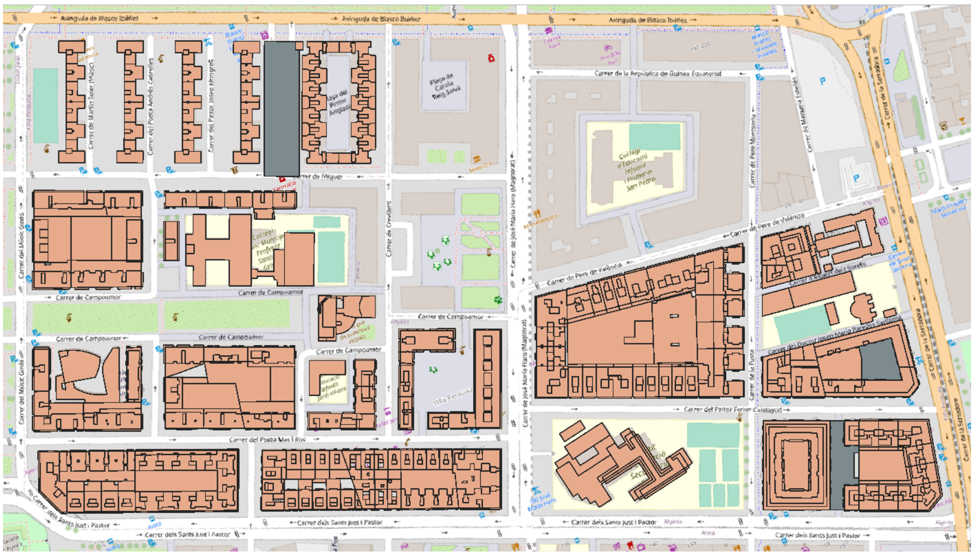


Figure 3.2: Base layer for building area calculations with QGIS. A street map of the neighbourhood is shown and the buildings' ground plan is overlapped onto it. The buildings with floors above the ground level are drawn in light brown. The features shown in grey are subterranean spaces, excluded in the calculation. This figure is a screen capture from QGIS 3.16.16 (Open Source Geospatial Foundation 2020).

- **Underground constructions.** Some features represent underground spaces, such as below-ground parking. To avoid including these features, a filter is applied, as seen in figure 3.2

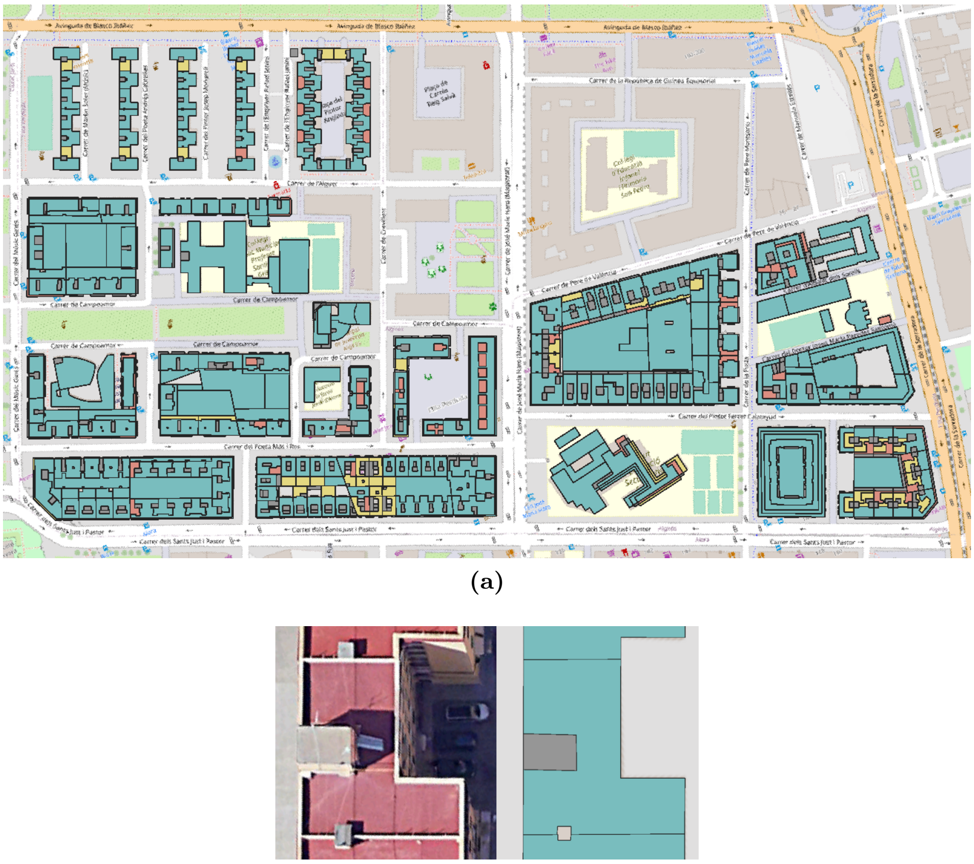
  `numberOfFloorsAboveGround > 0.` ,

- **Minimum area.** Even though in some technical guides there is no mention of a minimum required area for GR installation, not every roof will be greened. To address the benefits of greening bigger surfaces, a minimum area of 100 m^2 is required, following the criterium of the study performed in Lisbon (see 2.3) (Santos, Tenedório and Gonçalves 2016). Figure 3.3a shows the selected buildings as well as buildings with areas ranging between 75-100 m^2, 50-75 m^2, and below 50 metre^2. This area limitation serves also to exclude the roofs of the building access to roofs, which do not have the structural capacity required (as seen in 3.3b) and some smaller roof parcels. The applied filter is

  `area > 100.`

- **One floor buildings.** Single-storey building parts usually are used as terraces or communal inner courtyards. They could be used for green roofing, but the scenario in which they cannot be greened should be considered. Thus, two scenarios will be considered: one considering every building that fits the other requirements and another considering only multiple-storey buildings and single-storey building parts with more than 500 m^2 of area, to include large inaccessible roofs that can be greened. Both types of roofs can be seen in figure 3.4. The result of the excluded buildings based on this criterium is shown in figure 3.5. The filter used to identify the smaller one-storey roof is

  `numberOfFloorsAboveGround = 1 AND area < 500.`



(a)

(b)

Figure 3.3: Building layer for area calculations with QGIS. (a) A street map of the neighbourhood is shown and the above-ground buildings' plan is overlapped onto it. In light greenish-blue, the buildings with area higher than 100 m^2 are shown. In yellow, the buildings with areas between 75 m^2 and 100 m^2. In brown, the buildings with areas between 50 m^2 and 75 m^2. And in grey, the buildings with areas below 50 m^2. This figure is a screen capture from QGIS 3.16.16 (Open Source Geospatial Foundation 2020). (b) A zoom-in of one of the excluded areas is shown both in a QGIS (Open Source Geospatial Foundation 2020) (right) and a Google Earth Pro (LLC 2023) screen capture (left).

- **Slope.** Slope calculation is based on LiDAR height data. LiDAR data is not very accurate. The available Spanish data has a point cloud density of 0.5 pt m^{-1}, which is quite accurate for general purposes but cannot properly describe roof obstacles. When calculating the slope directly using LiDAR data cropped to the building layer (using the QGIS slope calculator algorithm), lots of high-value slope cells appear due to sharp changes in height associated with storey changes through an extended building, storage rooms on the roof or roof accesses. If the point cloud density were to be higher, these high values would occupy a decreasing portion of the roof area. However, given the current resolution, about 40% of the cells have values of slope higher than 30°, when the great majority of roofs in the neighbourhood are flat.

Figure 3.4: One of *L'Illa Perduda* blocks. In this block, a large one-storey grey roof can be seen at the centre of the image, surrounded at its bottom and top by smaller one-storey roofs used as terraces. This figure is an image produced with Google Earth Pro 7.3.6 (LLC 2023).

To address the slope limit of 10° for extensive roofing, an estimation of the mean slope of a building roof is made. First, only cells with a maximum slope of 30° are considered to exclude the abnormally high slope results. Then both the mean and the median slope are calculated for each building part. Then the mean to median normalised difference is calculated as

`NormalisedDif = abs(mean-median)/median` .

The median is considered to be a better statistic for slope since there are still abnormal values within the kept cells, and the median tends to ignore distribution outliers (if their number is kept proportionally small). Thus, two filters are established to exclude a roof: first, a median slope higher than 10°, and second, a mean to media normalised difference less than 1 (about a 75 percentile) is required. This is done to not exclude roofs that have higher abnormal values, which would separate the mean slope high above the median slope. The value set as a limit is indeed experimental since it works well for the considered buildings. A more in-depth study should be performed for the correct value of this parameter with a greater statistic. The established filter is then

`medianSlope < 10 OR NormalisedDif > 1` .

Establishing this filter excludes all traditional gable roofs and most of the rest of the pitched roofs, as well as some of the smaller flat roofs that are used as terraces (and thus are full of obstacles). The result of the slope filter is shown in figure 3.5.

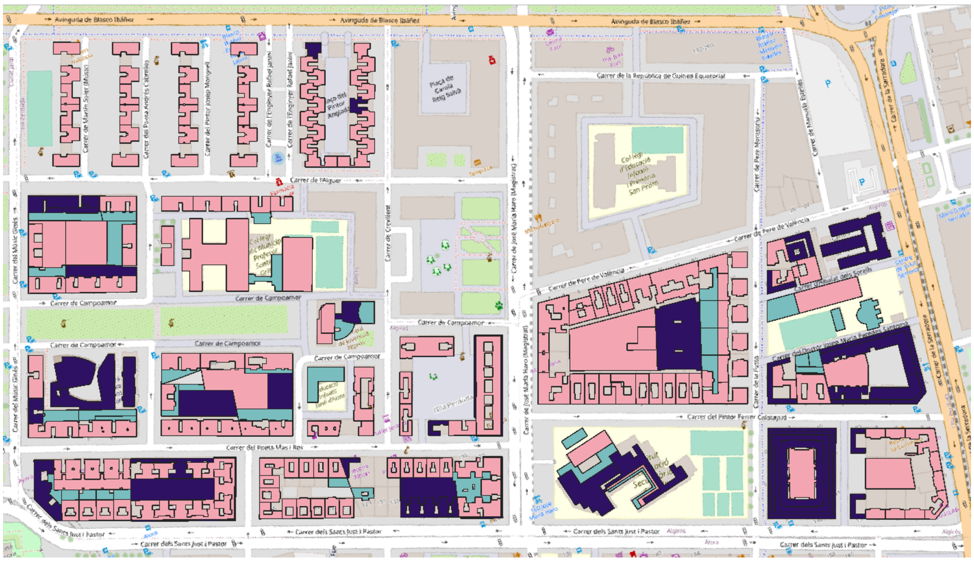


Figure 3.5: Final layer for building area calculations. A street map of the neighbourhood is shown and the buildings' ground plan is overlapped onto it. The buildings drawn in light pink are the accepted buildings for GR installation. In greenish blue, the one-storey buildings with an area less than 500 m^2 are shown. In dark blue, the buildings excluded based on the slope criteria are marked. This figure is a screen capture from QGIS 3.16.16 (Open Source Geospatial Foundation 2020).

- **Structural capacity.** Structural capacity is a determinant property for GR installation. Unfortunately, there is no data available on building roofs' load-bearing capacity. In some studies, the structural capacity is linked to the construction year as an estimative measure (Silva, Flores-Colen and Antunes 2017), based on local regulations on building requirements. A more detailed study of Valencian buildings could yield a relation between construction year and structural capacity, but in the present study, structural capacity will not be considered. This exclusion is one of the limitations of the presented methodology. Even if structural capacity cannot be assessed through GIS analysis, it can be partially considered by AV analysis, since non-suitable roofs such as sandwich panel roofs can be disregarded (see 4.3).

4. *Calculate the total area.* Finally, the total available area can be easily evaluated by adding up the surface areas of all selected buildings.

Artificial-vision-based calculation

The artificial-vision-based analysis is performed by processing aerial images of the neighbourhood. Thirteen different images were utilised to cover the whole neighbourhood, focusing on different building groups. The AV algorithms are based, on a first approach, on pixel-based segmentation, plus a human-supervised blob exclusion phase. The AV image processing aims to classify different building roofing solutions present in the neighbourhood throughout the different images, by also considering different regions of the pixel colour space that are linked to a specific roof typology. Here, the general proceedings will be detailed and, as an example, the segmentation process will be demonstrated for one of the roof types, the "red flat roof". The steps are the following:

1. *Produce the images.* Using Google Earth Pro (LLC 2023), a set of thirteen images was produced. For taking the images, all layers were set off, and an image of each building block was taken at 1080 HD resolution and saved as a PNG file.
2. *Crop and add the resolution scale data to the images.* The images are then preprocessed using Fiji (Schindelin et al. 2012), an open-source image processing package based on ImageJ2. First, they are cropped so that the building blocks occupy as much of the image as possible, to avoid detecting street features as building roofs. Second, the scale information is added to the image by using the scale bar that GEP adds to the pictures and the images are saved as TIFF (Tagged Image File Format) files.
3. *Image processing on* MATLAB. The image processing is based on two parallel proceedings: the roof type detection and classification and the actual image processing. The roof typology information is stored in a separate structure, while every image is processed in its own structure to produce a roof mask.

   The roofing features are segmented via pixel-based detection. For each roof type, a set of regions in the colour space are delimited, and pixels within those regions are initially classified as of the considered typology. The structure containing the **different roof typologies** is updated as new images are analysed. When a picture is processed with a new roofing type, the flow is the following (shown in figure 3.6):

   (i) **Roof type definition.** First, the new roof type is manually defined, and a name is given.

   (ii) **Colour region difinition.** Second, a set of different colour regions is defined (e.g. lighted and shadowed areas of the same roof, different colour coatings for flat roofs), and a set of pixels from the image are manually selected for each colour region.

   (iii) **Colour region delimitation.** Based on the selected pixels, a region of the colour space is associated with each colour region afore-defined. This delimitation is based on the statistics of the pixel values (maximum and minimum or mean and standard deviation) in two different colour spaces (RGB and HSV).

   (iv) **Colour region updating.** While posterior images are segmented, the pixel values of the different areas associated with each colour region are added to the pixel value collection, to improve the colour region delimitation. Unfortunately, this has not yet been implemented.

   Once the roof types that appear in a picture are correctly defined, the **image processing** in MATLAB is done in four stages:

   (i) **Image preprocessing.** First, the images are imported and preprocessed using a median filter, in which each pixel value is rewritten by the median of the pixel values within a 5x5 neighbourhood surrounding the target pixel. This filter easies the image segmentation but does not blur the edges too much.

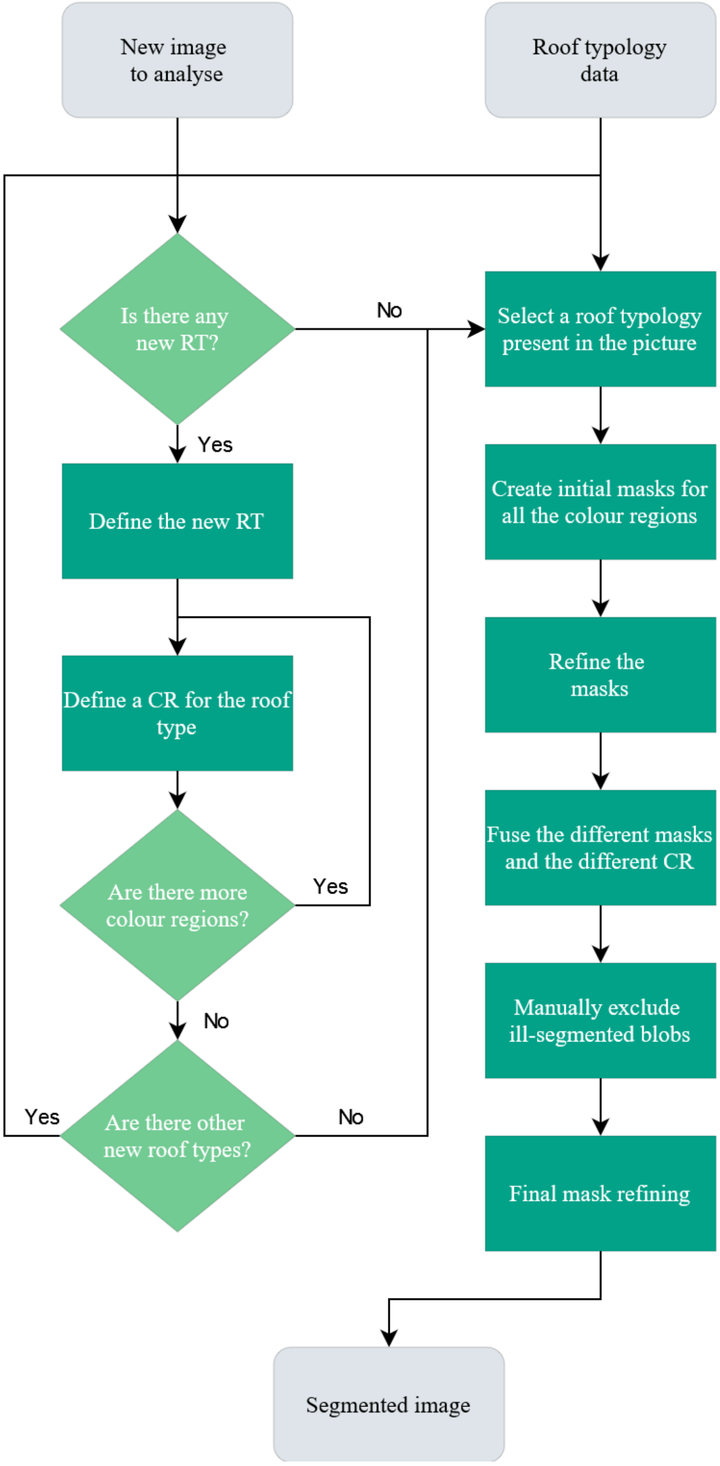


Figure 3.6: Flow diagram for image processing methodology.

Figure 3.7: Initial mask for roof detection. The original image is shown (left) along the resulting mask (right). The mask aims to detect the flat roof RT, the CR is associated with the illuminated, more red roofs. The CR is delimited in HSV space, using the mean of the pixel values and a 'cube' of side six times its standard deviation. As can be seen, this mask does not select shadowed pixels. The left image is the cropped result of the GEP (LLC 2023) picture (2020) and the image on the right is produced using MATLAB (MathWorks 2022).

(ii) **Initial mask creation.** Based on the colour region delimitation for each roof type (RT) present in the picture, a first mask is made for each colour region (CR) simply by selecting the pixels that correspond to the associated region of the colour space. Figure 3.7 shows the result of the initial mask in one of the neighbourhood pictures. In order to not select pixels outside the desired features, CRs must be chosen as strictly as possible. In this case, the selected colour region corresponds only to the saturated red roofs, and only to the illuminated areas.

(iii) **Mask filtering and refining.** Then, morphological operations are performed on the black-and-white (BW) mask to filter and refine the mask. As an example, in the case of 'flat roofs', all CRs are filtered following the same pattern. First, the mask holes that are below a certain area threshold are filled, to ensure that blobs with random holes are filled while blobs that should be separate but are randomly connected do not become fully fused. Second, a morphological mask opening is performed using a four-neighbour kernel by twice eroding and twice dilating the BW image. Third, blobs are classified by area, and those smaller than a certain threshold are erased. Then, a morphological mask closing is performed using a four-neighbour kernel by twice dilating and twice eroding the blob mask and the image is again filled with a larger threshold, hoping to fuse blobs that pertain to the same feature. Later, blobs are again thresholded based on area, now with a less tight maximum area. Finally, a last morphological closing is performed, this time by applying four times a dilation and an erosion operation, also using a four-neighbour kernel, and the image is again filled with a larger threshold. The result is shown in figure 3.8.

Figure 3.8: Filtered mask for roof detection. The original image is shown (left) along the resulting mask (right). The improvement from figure 3.7 is clear in cleaning selected points outside the building area and in polishing the detected surfaces. The left image is the cropped result of the GEP (LLC 2023) picture (2020) and the image on the right is produced using MATLAB (MathWorks 2022).

Figure 3.9: Fused mask for roof detection. The original image is shown (left) along the resulting mask (right). The mask aims to detect the flat roof RT, all the CRs have already been fused. Some blobs that do not belong to the roofs of the buildings have been selected but most of them will be erased in the next step. The left image is the cropped result of the GEP (LLC 2023) picture (2020) and the image on the right is produced using MATLAB (MathWorks 2022).

(iv) **Mask and CR fusing.** For each CR, four different masks–which will be called channels–are created and filtered, depending on whether the RGB or HSV colour space was used and the statistics used for defining the region of the colour space. These channels are fused by using logical operators directly between the masks or between the blobs that each mask detects. The different CRs should also be fused by performing a logical OR between all of them so that features detected from all CRs are considered. In the case of the 'flat roof', first, HSV and RGB channels are fused using what has been called a blob OR operation. This is, any blobs that share any pixels between both channels are fused, while blobs that are only detected by one channel are erased. Then, all illuminated CRs are fused using the OR operation and both different statistics channels (CR based on maximum and minimum pixel values and based on mean and standard deviation) are finally combined using a blob OR operation. Shadowed CRs must be treated differently, since RGB channels greatly overselect pixels. Thus, RGB channels are directly discarded and HSV masks are simply fused using an OR operation both in different statistics channels and different CR. Then illuminated and shadowed masks are fused using a logical OR operation. The results of this step are shown in figure 3.9.

Figure 3.10: Manually cropped mask for roof detection. The original image is shown (left) along the resulting mask (right). There is a part of the building's facade in the left bottom part which is still included in the mask, but this part is visually undifferentiable from the roof, so, naturally, the AV algorithm fails. The left image is the cropped result of the GEP (LLC 2023) picture (2020) and the image on the right is produced using MATLAB (MathWorks 2022).

Figure 3.11: Final mask for roof detection. The original image is shown (left) along the resulting mask (right). The left image is the cropped result of the GEP (LLC 2023) picture (2020) and the image on the right is produced using MATLAB (MathWorks 2022).

(v) **Manual blob rejection.** The next step consists of a manual blob rejection. Some of the detected blobs are not part of the roofs of the buildings and should be deleted. This step could be avoided if the images were to be georeferenced, since then the mask could be compared with the buildings' ground plans and exclude the blobs that do not lie within them. The results of the manual blob rejection for the example picture are shown in figure 3.10.

(vi) **Blob fusing.** Then, the mask must be once more filtered, since the light and shadow parts of the same roof need to be fused. To do so, the BW image is morphologically closed using an eight-neighbour kernel and three dilation and erosion operations, filled (using a large threshold), and, finally, individual blobs are morphologically closed (eight-neighbour, four operations). The final mask for the example picture is shown in figure 3.11, and an overlapped version is shown in figure 3.12.

4. *Available area calculation.* Finally, the total area is calculated by adding up the area of each blob from each picture.

Figure 3.12: Final mask for roof detection. The original image is shown, and the resulting mask is overlapped onto it in a blueish colour. The base image is the cropped result of the GEP (LLC 2023) picture (2020) and the overlapped mask is produced using MATLAB (MathWorks 2022).

3.3 Green roof potential calculation

The benefits calculation for GR installation will comprise the direct carbon sequestration evaluation, based on published experimental results; the cooling-and-heating-related energy savings, also based on published experimental results, and its associated carbon emission reduction; as well as a simple economic analysis.

Direct carbon sequestration

The literature on GR direct carbon sequestration is not extensive, and there are few experimental studies. A review from the *City University of Hong Kong* includes experimental studies on direct carbon sequestration performed before 2020 (Shafique, Xue and Luo 2020). One of the featured studies, from Murcian Centre of Edaphology and Applied Biology of the Segura river (CEBAS), used *Silene vulgaris* and *Lagurus ovatus* under irrigation at 40% of the measured evotranspiration ($530\,\mathrm{L\,m^{-2}\,yr}$) (Ondoño, Martínez-Sánchez and Moreno 2016). The experimental setups yielded a carbon sequestration of $4.4\,\mathrm{kgC\,m^{-2}\,yr^{-1}}$ and $1.9\,\mathrm{kgC\,m^{-2}\,yr^{-1}}$ respectively. As Murcia and Valencia share a similar climate, a value of $m_{\mathrm{C,D}} = 2\,\mathrm{kgC\,m^{-2}\,yr^{-1}}$ for direct carbon mass sequestrated per area and per year will be assumed for EGR. Thus, the total annual carbon sequestration potential will be evaluated as

$$M_{\mathrm{C,D}} = m_{\mathrm{C,D}} S_{\mathrm{T}}, \tag{3.1}$$

where $M_{\mathrm{C,D}}$ is the total carbon mass annually sequestrated and S_{T} is the total calculated EGR surface.

Energy savings and non-emitted carbon

Estimating the energy savings associated with massive GR installation is not easy. To do so, the cooling and heating load reduction will be considered. The Department of Architectural Constructions of the Polytechnical University of Valencia (UPV) conducted a computational study changing the climatic conditions through different Spanish cities (Borràs et al. 2022). If their results are interpolated to Valencia climatic and rainfall zones (both linear interpolation or linear fitting yield similar results), a single-storey house should experience a reduction of 17% to 20% in energy consumption after EGR installation. These results are similar to those of another computational study (validated by experimental results) based on a building in Benaguasil, a town from the metropolitan area of Valencia (Peñalvo-López et al. 2020). In general, most of the results analysed in the literature review (see section 2.2) yield results between 6% to 32% for single-storey buildings. For the calculation of the energy savings, an energy reduction of 17% will be assumed for single-storey buildings in Valencia.

In the case of high-rise buildings, the total energy savings drastically drop as the roof-to-wall area ratio decreases. A study on an eight-storey building in Madrid yielded only a 1% reduction in energy savings (Saiz et al. 2006). This study features a cooling load peak reduction of 25% on the highest floor and 9%, 2% and 1% on the subsequent floors. As, during stationary analysis, the energy consumption should evolve linearly approximately, these proportions will be considered when addressing energy savings. Thus, in general, as the highest floor of a multiple-storey building would behave as a single-story house, an energy reduction of 17% will be considered for the last storey of any building. Then, following the aforementioned proportions, a reduction of 6%, 1.5% and 0.5% will be considered for the next three highest floors of any building.

To calculate the energy savings, the energy consumption per habitable area will be evaluated for each building, so that the energy consumption of any floor is the product of this value and the floor area. First, based on the number of dwellings n_{D} of the building, so that the total energy consumption is calculated by will be calculated as

$$E_{\mathrm{B}} = E_{\mathrm{D}} n_{\mathrm{D}}, \tag{3.2}$$

where E_D is the mean energy consumption of a single dwelling in Valencia.

Second, the gross floor area of the building in the neighbourhood will be evaluated, this is,

$$S_{\mathrm{B,GFA}} = \sum_{\substack{\text{Building} \\ \text{parts}}} S_{\mathrm{P}} n_{\mathrm{F,P}}, \tag{3.3}$$

where S_{P} is the building part ground plan surface, n_{F} is the number of floors above ground that has the building, and the summation is performed over all the parts of the building with a different number of floors.

Then, the energy consumption per habitable area is simply

$$E_{\mathrm{B,S}} = \frac{E_{\mathrm{B}}}{S_{\mathrm{B,GFA}}}. \tag{3.4}$$

Finally, the energy consumption of any floor of the building is

$$E_{\mathrm{F}} = E_{\mathrm{B,S}} S_{\mathrm{F}}, \tag{3.5}$$

where S_F is the area of the considered floor. Thus, the total energy savings (in absolute value) will be calculated as

$$\Delta E_{\mathrm{T}} = \sum_{\mathrm{Floors}} \chi_{\mathrm{F}} E_{\mathrm{F}}, \tag{3.6}$$

where χ_{F} is the coefficient of energy reduction (i.e. 0.17, 0.06, 0.015, 0.005 for the highest four floors of the building) and the summation is performed over all floors of all building parts. Note that direct calculation of the energy consumption per floor is not possible since the dwelling information is given for the whole building while there may be different parts of the building with different numbers of floors.

Then, the energy-related carbon reduction $M_{\mathrm{C,E}}$ would be obtained as

$$M_{\mathrm{C,E}} = m_{\mathrm{C,E}} \Delta E_{\mathrm{T}}, \tag{3.7}$$

where $m_{\mathrm{C,E}}$ is the emissions mass per energetic consumption, given the local energetic mix.

Chapter 4

Case study and results

Explain the limitations of the case study, why L'Illa Perduda and a possible extension to Valencia.

The case study for the application of the methodology that has been presented in this project is the L'Illa Perduda neighbourhood in the city of Valencia. The idea behind the application of this project's methodology to this small quarter is to serve as a probe for a future extension to the whole city of Valencia.

4.1 Scoping

The delimitation of the L'Illa Perduda neighbourhood is based on the administrative district boundaries. The extension of this methodology to the whole city would include 19 districts, divided into 87 neighbourhoods (*Oficina de estadística - Barrios 2023* 2022).

L'Illa Perduda

L'Illa Perduda is one of the small neighbourhoods of Valencia. It comprises an area of 23.2 ha, a 0.2% of the city's extension, and a population of 8474 inhabitants, a 1% of Valencia's population, with a population density of $365.3\,\mathrm{ha}^{-1}$, higher than the municipal mean density. Its construction dates back to 1962, as a continuation of the extension of Valencia toward the coastal towns, when the first buildings were constructed in the middle of the Valencian 'Huerta'. That is why the neighbourhood is called 'L'Illa Perduda', which translates to 'the lost island'. The site and location blueprints of the neighbourhood can be found in part III.

The choice of the L'Illa Perduda neighbourhood as the probe quarter for this project's methodology is due to the extensive research already done into the zone by the Institute of Energy Engineering (IIE) of the UPV.

Future extension to Valencia

The final idea for this project is to extend the calculation to the whole city of Valencia. Scaling up the methodology requires some proceedings to be improved:

- **Using georeferenced images.** To cover the L'Illa Perduda neighbourhood, thirteen images were needed. This could mean that about 7500 images would be needed to cover the entire city of Valencia (calculation based on the area of the city, the actual number would probably be smaller since L'Illa Perduda has a high building density). The manual obtention of the pictures could slow down the calculation. Working with georeferenced images could ease the task.

  The Spanish National Centre of Geographic Information offers a catalogue of georeferenced orthophotos that cover the entire country, with a mesh pitch of 0.25 m (*Centro de Descargas* 2023) (*Fotos e imágenes aéreas - Ortofoto PNOA máxima actualidad*). Eight of these orthophotos would cover the entire city. When using georeferenced images, a simple script could automatically produce close-up pictures of every building in the cadastral database, which could be easily handled and processed by MATLAB.

- **Improving image automatic processing.** If georeferenced images were to be used, the cadastral ground plan of every building could be used to only accept AV-segmented blobs that lie within the building, highly reducing or even removing the need for human supervision.

- **Enhancing segmentation performance.** Some studies suggest that OBIS performs better than pixel-based image analysis PBIS for unmanned aerial vehicle images (Sibaruddin et al. 2018). Algorithms like multiscale image segmentation could be used in combination with the already-used PBIS algorithms to yield even better results. Also, other techniques like mean shift clustering, k-means clustering or edge detection could be introduced into the main segmentation script. These algorithms have been experimented with, and they individually yield good results but need to be combined properly to give a definitive segmentation. They could not be included in the final script due to time constraints.

- **Connecting the AV area calculation with the energetic analysis.** Right now, as the aerial images are not enriched with geographic information, there is no way but manually to introduce the storey information so that the energetic calculation can be performed. The introduction of georeferenced images would allow for the automatic calculation of energy consumption reduction via AV-based analysis, simply by transferring the cadastral properties of the different buildings to the corresponding mask.

- **Using an open-source programming language.** In the future, choosing open-source options, such as Julia or Python, over MATLAB, would be preferable. Open-source and free software promotes collaboration and innovation, is free of charge and transparent, thrives based on community support, and improves education and STEAM training.

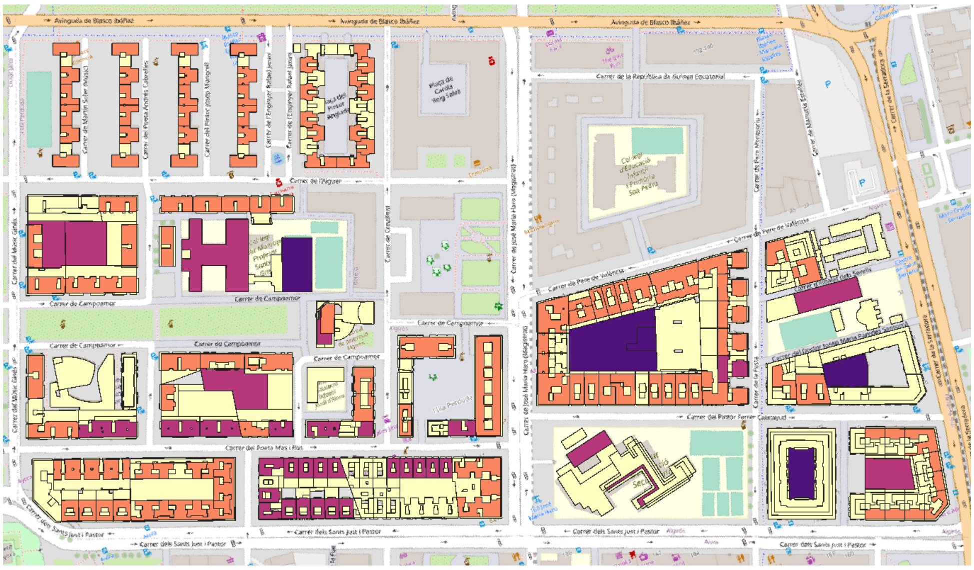


Figure 4.1: Final selection of buildings. A street map of the neighbourhood is shown and the buildings' ground plan is overlapped onto it. The buildings with floors above the ground level are drawn in cream (light yellow). The rest of the features are the selected roofs. They are classified based on their median height (from LiDAR data). Buildings in purple are less than 10 m high. Buildings ranging between 10 m and 25 m are shown in pink and buildings higher than 25 m are drawn in orange. This figure is a screen capture from QGIS 3.16.16 (Open Source Geospatial Foundation 2020).

4.2 Green roof potential calculation – GIS study

The results of the potential calculation for the L'Illa Perduda neighbourhood will be presented in this section.

Available area calculation

The original cadastral data includes 165 different buildings, separated into 1157 different parts. The total area of the original buildings' ground plan (nonprocessed cadastral data) adds up to 9.96 ha (counting only buildings that extend above the ground level, not considering underground areas), a 43% of the neighbourhood's extension.

The parts that extend over an area higher than 100 m^2 cover a surface of 7.93 ha, a 34% of L'Illa Perduda. When only the one-storey buildings with an area higher than 500 m^2 are included, the area is reduced to 7.10 ha, a 31% of the quarter. Then, as the slope criterium is considered, the available area drops down to 4.83 ha, a 21% of the neighbourhood's extension and a 48% of the original building area. Table 4.1 includes the area and percentage values of the accepted buildings with different filters applied, also specifying the results for residential buildings.

Most of the selected buildings are residential buildings, but some of them are classified as industrial, retail or public services. Also, not all the parts of residential buildings hold dwellings within them. Figure 4.2 shows the residential buildings with and without dwellings. The selected residential buildings extend over an area of 3.99 ha, a 40% of the total building area, of which only 0.09 ha correspond to building parts that hold no residences, while 0.84 ha correspond to not residential buildings, an 8% of the total building area.

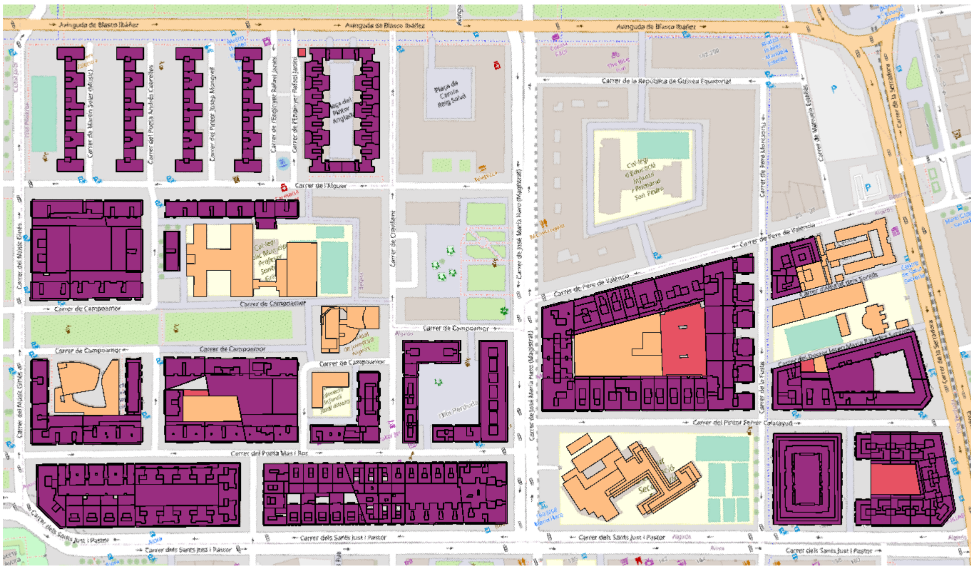


Figure 4.2: Buildings of L'Illa Perduda classified by use. A street map of the neighbourhood is shown and the buildings' ground plan is overlapped onto it. All the shown buildings rise above ground level. The buildings in purple are classified as residential buildings and hold dwellings within them. Buildings in red are residential buildings without residences, and buildings in orange are non-residential buildings. This figure is a screen capture from QGIS 3.16.16 (Open Source Geospatial Foundation 2020).

Table 4.1: Available rooftop areas after the building filtering process. The area is given in hectares (1 ha = 1 hm^2) and as the percentage of the neighbourhood area in brackets. The different filters applied to the buildings are: including only buildings with an area higher than 50 m^2, 75 m^2, and 100 m^2; including buildings with an area higher than 100 m^2 if they have multiple storeys and single-storey buildings (1SB) with areas higher than 500 m^2; and considering no slope criterium or the slope criterium explained in the methodology, using 30° or 50° as a limit to accept slope cells to compute the median slope in each roof (see section 3.2). The results are given for all building parts, and parts corresponding to residential buildings.

Current use	Slope filter	Building plan	Thresholds			
			Area >			Area > 100 m^2
		(ha (%))	50 m^2	75 m^2	100 m^2	1SB area > 500 m^2
All	No	9.96 (43)	8.79 (38)	8.41 (36)	7.93 (34)	7.10 (31)
	30°	6.06 (26)	5.73 (25)	5.56 (24)	5.33 (23)	4.83 (21)
	50°	4.77 (21)	4.66 (20)	4.61 (20)	4.47 (19)	4.06 (18)
Residential	No	8.12 (35)	7.01 (30)	6.66 (29)	6.25 (27)	5.68 (24)
	30°	4.98 (21)	4.67 (20)	4.52 (19)	4.29 (18)	3.99 (17)
	50°	3.75 (16)	3.65 (16)	3.61 (16)	3.47 (15)	3.25 (14)

The final area considered by this approach is 4.83 ha, about 50% of the original area. Half of the neighbourhood could be greened, lacking a structural study of the roof's load-bearing capacity. Of all this area, 1.87 ha, about 40% of the selected area is occupied by roofs that are less than 25 m high, which could improve green connectivity, following the study performed in Madrid (Velázquez et al. 2019), lacking a connectivity analysis. Also, the buildings that are less than 10 m high cover an area of 0.56 ha, about a 10% of the selected area, which could help reduce the UHI effect.

Direct carbon sequestration

Assuming EGR installation in all the available roofs of the neighbourhood, and using the value of $m_{\mathrm{C,D}} = 2\,\mathrm{kgC\,m^{-2}\,yr^{-1}} = 7.3\,\mathrm{kgCO_2\,m^{-2}\,yr^{-1}}$ for direct carbon removal, as explained in the methodology (see section 3.3), **total direct carbon sequestration** of

$$M_{\mathrm{C,D}} = 96.6\,\mathrm{tC\,yr^{-1}} = 354.2\,\mathrm{tCO_2\,yr^{-1}}$$

would be expected. Table 4.2 shows the annual direct carbon sequestration expected in each scenario.

Table 4.2: Carbon sequestration after building filtering process. The carbon mass sequestrated is given in tonnes per year ($\mathrm{tC\,yr^{-1}}$). The different filters applied to the buildings are: including only buildings with an area higher than $50\,\mathrm{m^2}$, $75\,\mathrm{m^2}$, and $100\,\mathrm{m^2}$; including buildings with an area higher than $100\,\mathrm{m^2}$ if they have multiple storeys and single-storey buildings (1SB) with areas higher than $500\,\mathrm{m^2}$; and considering no slope criterium or the slope criterium explained in the methodology, using 30° or 50° as a limit to accept slope cells to compute the median slope in each roof (see section 3.2).

		Building plan	Thresholds			
			Area >			Area > $100\,\mathrm{m^2}$
		($\mathrm{tCO_2\,yr^{-1}}$)	$50\,\mathrm{m^2}$	$75\,\mathrm{m^2}$	$100\,\mathrm{m^2}$	1SB area > $500\,\mathrm{m^2}$
Slope criterium	No	730.4	644.6	61.67	581.5	520.7
	30°	444.4	420.2	407.7	390.9	354.2
	50°	349.8	341.7	338.1	327.8	29.77
Residential	No	595.5	514.1	488.4	458.3	416.5
	30°	365.2	342.5	331.5	314.6	292.6
	50°	275.0	267.7	264.7	254.5	238.3

Energy savings and indirect carbon reduction

The total gross floor area of the neighbourhood is $S_{\mathrm{T.GFA}} = 56.93\,\mathrm{ha}$, which gives a mean of 6.32 floors per building (which is coherent with high-rise buildings of about 8 storeys and wide one-story parts). The total residential energy consumption, using the value of 0.719 toe as the mean energy consumption per dwelling in the Mediterranean part of Spain, based on the Episcope–Tabula Catalogue of Residential Building Typology of Spain (García-Prieto, Serrano and Ortega 2016), is $E_{\mathrm{T}} = 3017\,\mathrm{toe}$.

For the selected buildings, the total gross floor area is $S_{\mathrm{S,GFA}} = 29.93\,\mathrm{ha}$, of which 28.32 ha are residential surface, and 28.22 ha are dwelling floor. The selected buildings account for 1680 toe of energy consumption. Following the presented methodology, the **total residential energy savings** are

$$\Delta E_{\mathrm{T,R}} = 55.3\,\mathrm{toe},$$

which corresponds to 2% of the total energy consumption and 3% of the selected building's energy consumption.

In the Valencian Community, based on the results of the IVACE data report on 2020 (*Datos Energéticos de la Comunitat Valenciana (2020)* 2023), the CO_2 emissions per electric energy consumption are $m_{\mathrm{C,E}} = 154\,\mathrm{gCO_2\,kW^{-1}\,h^{-1}}$. If the energy consumed per dwelling were to be fully electric, the energy savings would produce a reduction in emissions (**indirect carbon reduction**) of

$$M_{\mathrm{C,E}} = 99.10\,\mathrm{tCO_2},$$

which yields a **total reduction of net atmospherical carbon** of

$$M_{\mathrm{C,T}} = 453.4\,\mathrm{tCO_2}.$$

In the worst possible scenario, the **energy consumption would be backed by burning natural gas**, which has a higher rate of emissions ($m_{\mathrm{C,NG}} = 202\,\mathrm{gCO_2\,kW^{-1}\,h^{-1}}$ (*Factores de emisión. Registro de huella de carbono, compensación y proyectos de absorción de dióxido de carbono.* 2023)). This yields a total mass of non-emitted carbon of

$$M_{\mathrm{C,NG}} = 129.9\,\mathrm{tCO_2}.$$

The benefits GRs provide rise proportionally when installed in **single-storey buildings**. They have a higher roof-to-wall-area ratio, improve green connectivity and lower ground-level temperatures. If we consider only the one-storey residential buildings that hold dwellings in them, they extend over an area of 0.36 ha and consume $15.5\,\mathrm{toe\,yr^{-1}}$. The energy savings on these buildings rise to $2.63\,\mathrm{toe\,yr^{-1}}$, which is 17% of the total energy they consume (as it was established before, the slight deviation of the calculation is due to precision errors). For multiple-storey buildings, which occupy an area of 3.54 ha, the consumption is 1664.5 toe and the energy savings are 52.7 toe, a 3% of the consumption.

There is no clear way to estimate energy consumption on **non-residential buildings**. To try and illustrate the benefits of GR installation in these edifices, an energy consumption per gross floor area of $59.70\,\mathrm{toe\,ha^{-1}}$ (the mean for residential buildings in the neighbourhood, whose gross floor area is $S_{\mathrm{R,GFA}} = 50.54\,\mathrm{ha}$) could be assumed, even though, in reality, it would probably be more. From the selected buildings, the non-residential rooftops cover an area of 0.84 ha, and assuming a total consumption of 96.16 toe, GR installation would provide energy savings of 10.24 toe, a 10.6% of the consumption. Table 4.3 shows a summary of the results for residential and non-residential buildings.

Table 4.3: Summary of area, energy, and carbon sequestration for residential and non-residential selected buildings, as well as the calculations for the sum of both buildings, also adding the buildings that hold no dwellings within them when reasonable.

	Residential			Non-residential	Total (+ND)
	MSB	SSB	Total		
S_{F} (ha)	3.54	0.36	3.90	0.84	4.74 (4.83)
S_{GFA} (ha)	27.87	0.36	28.22	1.61	29.84 (29.93)
Mean number of storeys	7.87	1.00	7.24	1.92	6.30 (6.20)
E_{C} (toe/yr)	1665	15.5	1680	96	1786
ΔE_{T} (toe/yr)	52.7	2.63	55.3	10.24	65.5
$M_{\mathrm{C,D}}$ (cotonne/yr)	259.6	26.40	286.0	61.6	347.6 (354.2)
$M_{\mathrm{C,E}}$ (cotonne/yr)	94.39	4.71	99.10	18.34	117.4
$M_{\mathrm{C,NG}}$ (cotonne/yr)	123.8	6.18	129.9	24.06	154.0

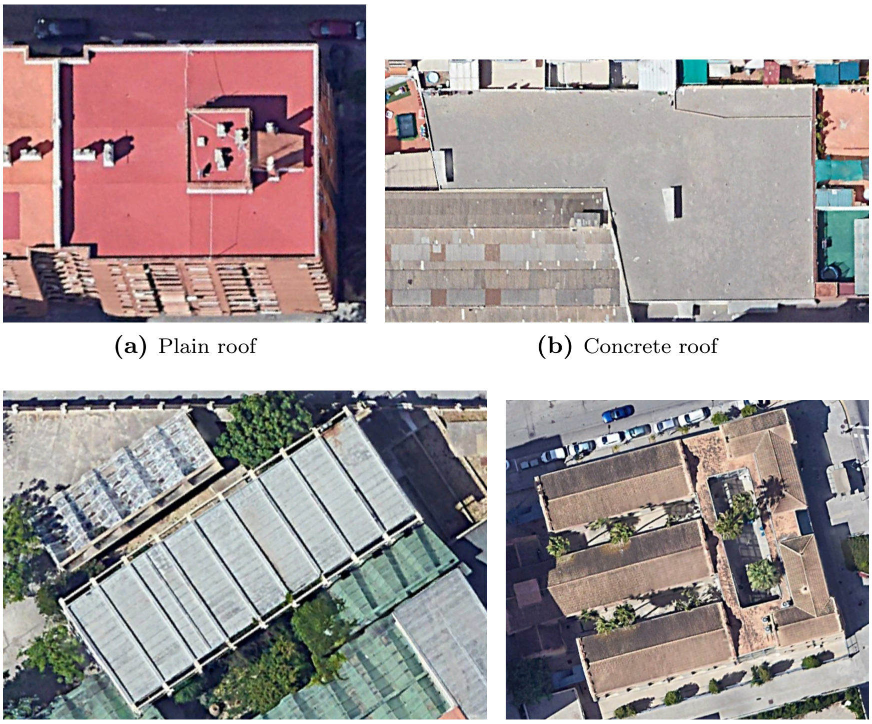
4.3 Area calculation using MATLAB

The results of the area calculation using AV algorithms in MATLAB will be presented in this section.

Roofing types in L'Illa Perduda

In the L'Illa Perduda neighbourhood, four different roof types can be found throughout the different buildings, all of them shown in figure 4.3.

- **Plain roofs.** Typical insulated roof of Valencia. Most of them are painted in various shades of red, but some of them are painted in blue, white or grey. Figure 4.3a.
- **Concrete roofs.** Large concrete roofs that are used in industrial buildings within residential areas. Figure 4.3b.
- **Sandwich panel roofs.** Light-weight prefabricated roofs that are mostly used in industrial areas. In the neighbourhood, the sandwich panel roofs belong to a public high school and a modern church. Figure 4.3c.
- **Gable roofs.** Most of the gable roofs of Valencia are made of brownish tiles, which makes them easily recognisable. Figure 4.3d.


(a) Plain roof **(b)** Concrete roof

(c) Sandwich pannel roof **(d)** Gable roof

Figure 4.3: Aerial imaging examples of different roofs found in the L'Illa Perduda neighbourhood. These photos were produced using Google Earth Pro (LLC 2023).

Both gable roofs and sandwich panel roofs are not suitable for greening. The exclusion of the former is because of their slopes being, in general, higher than what is accepted to install a GR without extra supporting structures (see section 2.2 for slope requirements and 5.1 for the GIS classification of these roofs as unsuitable). The exclusion of the latter is because it lacks the structural capacity to support the weight of a GR. Concrete roofs could be suitable for greening, but they are few, for industrial purposes, and they are difficult to isolate using pixel-based image segmentation (PBIS) techniques, such as the ones that have been implemented. The detection of concrete roofs could be done using object-based image segmentation (OBIS) techniques, and gable and sandwich panel roofs could be done using pattern recognition techniques (even though gable roofs share a quite similar region of the colour space to be recognisable by OBIS).

Plain roofs and colour regions

Plain roofs are easily recognisable. They are coloured in a mostly pure colour that contrasts the streets, and most of the time separated by white parapets. Thus, the first algorithm developed to detect rooftop available areas has been used to identify this roofing type. To do so, the roof was divided into seven different colour regions, depicted in figure 4.4. Red and orange roofs were present in twelve of the thirteen analysed images, while dark blue and wine colours were only used for one image each.

(a) Red (light and shadow) **(b)** Orange (light and shadow)

(c) Dark blue (light) **(d)** Wine (light and shadow)

Figure 4.4: Aerial imaging examples of the different colour regions delimited in the L'Illa Perduda neighbourhood for plain roofs. All colours except for dark blue have two regions associated, one concerning the illuminated parts and one concerning the shadows. In the case of the dark blue roofs, the shadows were not considered since the streets were mistakenly selected as part of the shadowed dark blue colour region. These photos were produced using Google Earth Pro (LLC 2023).

In general, shadowed areas are difficult to recognise, thus reducing the area of the masked blobs, especially in small shadows that tend to be seen as darker, such as those of the parapets (figure 4.5a) or from objects with pointy shapes (figure 4.5b). Also, temporary obstacles that are in the moment that the photo was taken, such as hanging laundry to be dried (figure 4.5c), hinder the recognition. Moreover, if the floor is patterned or has drastic variations in colour (figure 4.5d), it will not be correctly masked.

Table 4.4: Summary of the masked areas applying the AV algorithm to plain roofs, applying no threshold and thresholding the area of the masked blobs. The values are given in ha and as a percentage of the total residential buildings once the corresponding threshold is applied and as a percentage of the selected buildings under the corresponding threshold and the slope criterium (in brackets, separated by a comma and in that order).

No threshold	Blob area threshold		
(ha (%,%))	$> 50\,\mathrm{m}^2$	$> 75\,\mathrm{m}^2$	$> 100\,\mathrm{m}^2$
3.89 (48, 78)	3.49 (50, 75)	3.26 (49,72)	3.05 (49,71)

(a) Parapet shadow (left limit)
(b) Antenna shadow

(c) Obstacles (hanging laundry)
(d) Patterned floor

Figure 4.5: Aerial imaging examples of the different obstacles to rooftop segmentation. The images are shown, and the resulting masks are overlapped onto them in a blueish colour. These photos were produced using Google Earth Pro (LLC 2023).

Area results

Once the masks are produced and filtered, the resulting blobs are thresholded depending on their area, and the total area of the mask is summed over the different pictures. The results of the area calculation are summarised in table 4.4. The selected roof type is exclusively found in residential buildings. The resulting masks select plain areas and avoid permanent (and temporal) obstacles. Therefore, these results should be compared with the GIS results once the slope criterium is applied. Also, the comparison should be more fair when smaller areas are thresholded.

Chapter 5

Discussion of the results

The methodology has been successfully applied to the L'Ill Perduda neighbourhood. The results highlight that about 50% of the roofs of the neighbourhood could be used for GR installation. The direct benefits calculation of the GR retrofit is not flattering but should be considered in perspective.

5.1 Available area

The total available area for GR installation calculated for the L'Illa Perduda neighbourhood is 4.83 ha, 48% of the total neighbourhood rooftop area. The selected residential buildings extend over an area of 3.99 ha, 49% of the total residential rooftop area, and the selected non-residential rooftops account for 46% of the area, 0.84 ha. The total rooftop area of the whole city of Valencia is 1607 ha, and 1086 ha correspond to residential roofs. Assuming only residential roofs can be greened and that the same proportion of rooftops are adequate for greening, an area of

$$S_{\mathrm{R,V}} = 499.6\,\mathrm{ha}$$

could be used for **roof greening in the city of Valencia**.

Goodness of slope indicators

The slope criteria of only accepting roofs with a slope less than 10° is difficult to apply due to the already-explained abnormally high values that appear when slope calculation is performed (see 3). To try and understand the problem and study the possible indicators, a simple study of the different parameters will be made below.

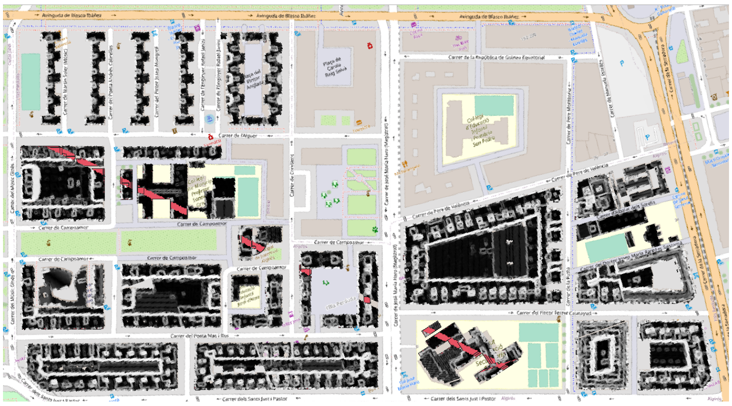
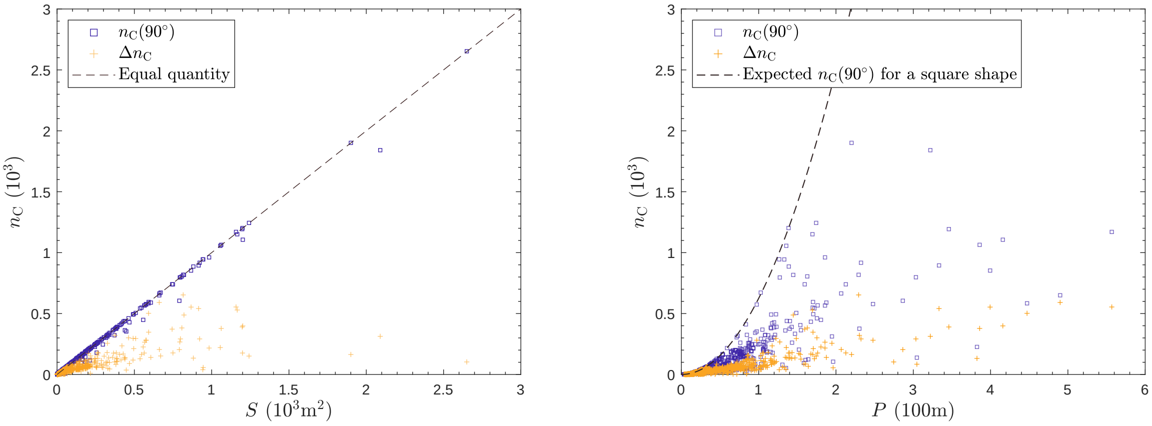


Figure 5.1: Slope raster layer of the buildings in the L'Illa Perduda neighbourhood. A street map is shown and overlapped onto it, the buildings are shown in red and the raster above it, lighter cells mean more pronounced slopes (ranges from 0 to 88°). The limits between areas at different heights are marked as highly pitched areas. A band without data can be seen crossing the neighbourhood from the middle left to the bottom right. This image is a screenshot from QGIS (Open Source Geospatial Foundation 2020)

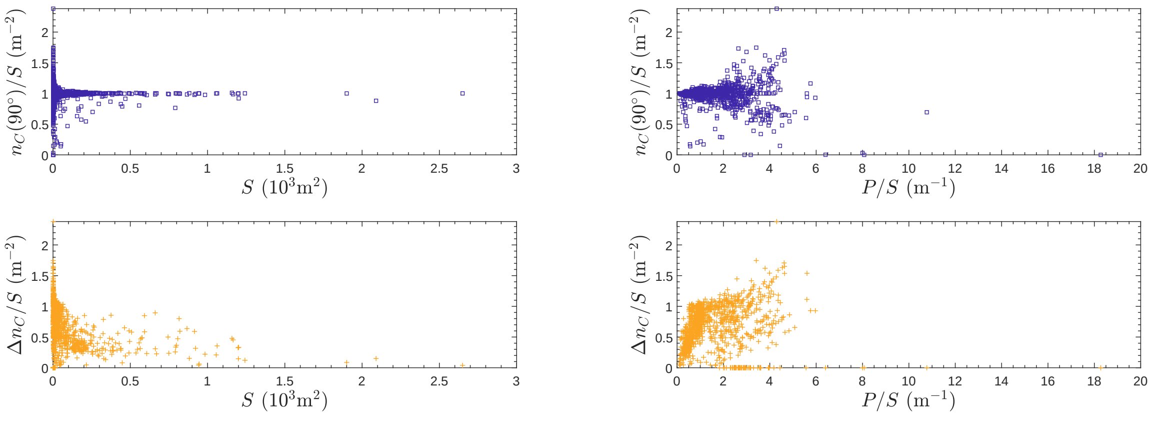


(a)

(b)

Figure 5.2: Scatter plot of original ($n_C(90°)$) and rejected (Δn_C) slope cell quantity within the rooftop limits versus area and perimeter of each building part. For areas below $100\,\text{m}^2$, the original cell density $n_C(90°)/S$ moves away from the unit value as is expected. The rejected cell density $\Delta n_C/S$ reduces as the area grows. The original cell density tends to disperse as the P/S ratio increases, while the rejected cell density always increases with the ratio. These figures have been produced using Matlab (MathWorks 2022).

When calculating the slope, a raster layer with height data of cell size of $1\,\text{m}^2$ cropped down to the limits of the buildings' rooftops of the neighbourhood was used. The slope raster originally had $n_{C,T}(90°) = 97768$ cells within the buildings limits, which would correspond to an area of 9.78 ha, a bit less than the total rooftop area (9.96 ha) since an 8 m wide line without data crossed the neighbourhood. Figure 5.1 shows the raster layer of slope cells. When using a maximum value of 30° to left out abnormal values, only $n_{C,T}(30°) = 55518$ cells were kept (57% of the original quantity). This huge reduction can be a cause of error in the slope criteria application, but in the next section, it will be seen that it is not the case if the correct indicators are selected.



(a) **(b)**

Figure 5.3: Scatter plot of original ($n_\mathrm{C}(90°)/S$) and rejected ($\Delta n_\mathrm{C}/S$) slope cell over quantity within the rooftop limits versus surface area and the perimeter-to-surface ratio of each building part. There is a clear linear relation between the total cells and the rooftop area, being close to an identity function, which would be expected since the cell size is $1\,\mathrm{m}^2$. The rejected cells grow with the area for smaller rooftops but for surfaces above $1000\,\mathrm{m}^2$ this growth stops, even seeming to decrease (but the number of samples is quite low for bigger surfaces). In the second scatter, the number of rejected cells seems to keep growing with the perimeter. The number of original cells also grows with the perimeter but in a much more dispersed way. These figures have been produced using MATLAB (MathWorks 2022).

Figure 5.2 shows the relation between cell counts, n_C, and rooftop area, S, or perimeter P. In figure 5.2a, original cell counts ($n_\mathrm{C}(90°)$, as including all cells is equivalent to using 90° as a maximum) grow linearly with S and are a nearly equivalent measure of rooftop area. The rejected cell ($\Delta n_\mathrm{C} = n_\mathrm{C}(90°) - n_\mathrm{C}(30°)$) growth stops and even decreases for larger areas. Figure 5.2b shows how rejected cells keep growing as the perimeter increases. This shows that $n_\mathrm{C}(90°)$ is more correlated with the area and that Δn_C is more correlated with the perimeter.

An indicator of how much rejecting cells can affect the result is the quotient of the number of rejected cells and the surface, as it can be interpreted as the portion of the surface that is rejected. Also, shapes with larger perimeters could have more rejected cells than others with the same area, so the perimeter-to-area ratio should also be considered. Figure 5.3 shows these cell-count-to-surface ratios ($n_C(90°)/S$ and $\Delta n_C/S$) scatterd depending on the area S and on the perimeter-to-area ratio P/S. The first scatter plot in figure 5.3a can be surprising. Most of the points show the already-mentioned direct relation between original cell counts, $n_C(90°)$, and area, S, but for small surfaces, the ratio $n_C(90°)$ ranges between 0 to 2 (or even 2.5). This is due to the counting algorithm used by QGIS and the missing line of data. When the ratio is considered versus the rooftop area, the portion of the rejected surface tends to decrease, since bigger shapes tend not to be affected by border effects such as abnormal values. Figure 5.3b shows how only in low P/S roofs, the portion of cell counts tends to 1, except for some few rooftops, that lie within the no-data band, and have low cell covering. The second plot shows the rejected cells to rooftop area ratio and how it increases, in general, with the perimeter-to-area ratio. There seem to be two different parts, one where the growth is much faster and one where it keeps growing, but slowly.

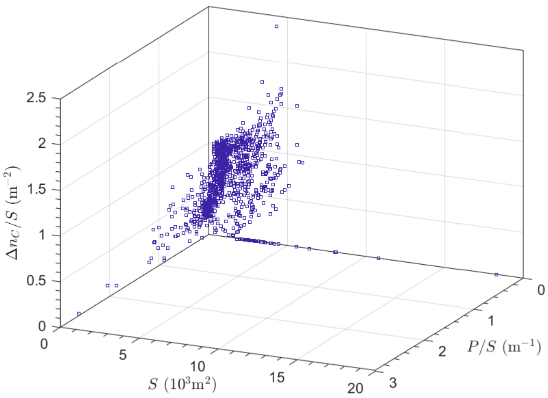


Figure 5.4: Three-dimensional scatter plot of the neighbourhood rooftops by surface S, perimeter-to-surface ratio P/S, and rejected cell density $\Delta n_C/S$. The points lie mainly close to the $S = 0$ and $P/S = 0$ surfaces. It can be seen that points with high P/S have low values of S and are have also the highest values of $\Delta n_C/S$. This plot was produced using MATLAB (MathWorks 2022).

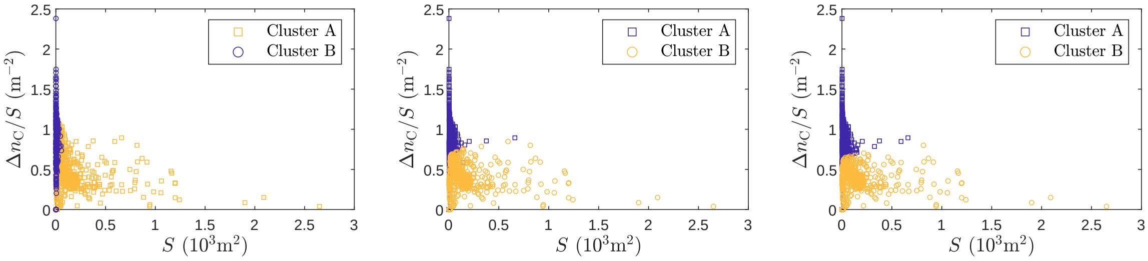
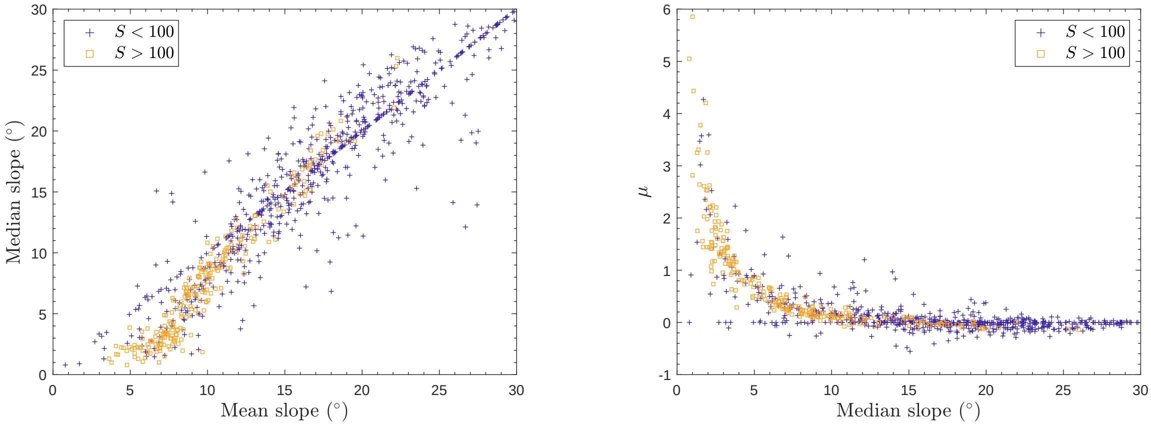


Figure 5.5: Scatter plots of $\Delta n_C/S$ vs. S, separated based on the results of the k-means clutering with $k = 2$ and scaling factors $(10^{-3}, 0.3, 1)$, $(10^{-3}, 0.275, 1)$ and $(10^{-3}, 0.25, 1)$ for the 3-tuple $(S, P/S, \Delta n_C/S)$. Three figures can be cut at S values from $50\,\mathrm{m}^2$ to $150\,\mathrm{m}^2$ to isolate mainly points from cluster B with larger areas. These plots were produced using MATLAB (MathWorks 2022).

Figure 5.4 shows a three-dimensional scatter plot of the results of figure 5.3. The points spread close to the surfaces $S = 0$ and $P/S = 0$. The two different distributions that were found in figure 5.3b are now clearly identified as small rooftops, which reach higher values of rejected cell density and perimeter-to-surface ratio, and the rest, with lower values that decrease as S increases. If this three-dimensional space is thought of as a vector space, clustering algorithms could separate both distributions. Using k-means clustering with $k = 2$, varying the scaling for different axes (as they are different magnitudes is difficult to put them together to form a balanced vectorial space), a cut in the surface axis can be made to isolate the larger area distribution, between $50\,\mathrm{m}^2$ to $150\,\mathrm{m}^2$ depending on the scaling factors, as shown in figure 5.5.



(a) **(b)**

Figure 5.6: Scatter plots of the mean and median slope of the buildings' roofs, and the normalised difference between those two, μ, separated based on rooftop area. The mean and median tend to have a linear but more dispersed relation for higher slopes, while less accused slopes tend to have a lower median slope than their mean slope, a product of the abnormal values. The μ parameter and the median slope show a nice correlation, with higher values of the normalised difference corresponding to lower values of the median slope. This could mean that higher values of μ could be associated with flat or less pitched roofs that are affected by abnormal values. These figures have been produced using MATLAB (MathWorks 2022).

Finally, it is necessary to analyse the indicators of slope. Figure 5.6 show the scatter plots of the mean and median slope and its difference divided by the median value, μ. The effect of abnormal values, when limited (this is, when) In smaller rooftops, abnormal values have a larger effect, drawing the median slope closer to the mean, or even higher, while larger roofs tend to have lower values of slope, and a median slope lower than the mean since the abnormal values tend to rise the mean value. When the normalised difference μ is compared to the median slope, a clear relation can be seen, where higher values of μ are associated with lower values of the median slope. Thus, it makes sense to set a minimum value for the parameter μ, apart from a maximum to the median slope (which should include more roofs than the same maximum to the mean slope). A minimum value of $\mu = 1$ will not include many rooftops that are not included by the maximum value of 10° for the median slope, so maybe a lower value could help reduce the error. It has been decided to keep it at $\mu = 1$ for this report, but lower values should be tested in the future. In any case, the goodness of the limits will be discussed in the next section.

Pitched roofs and terraces selection goodness

To check the goodness of the slope criterium, individual building parts not correctly accepted or rejected have been manually detected along the whole neighbourhood. The manual criteria for accepting or rejecting a building based on 'slope considerations' would be to reject a rooftop whether (1) the roof is clearly pitched (e.g. gable roofs, note that most of the rooftops in the city are flat due to the absence of snow and minimum rainfall) or (2) the roof is used as a terrace and there are different obstacles on it.

After the manual revision, a total of 168 (15% of the building parts) rooftops have been misselected by the slope criteria, a total area of 0.88 ha (9% of the total rooftop area). Most of the rooftops have a small area, 147 of them below 100 m^2 (0.42 ha) and 128 below 50 m (0.28 ha). The largest (0.11 ha) misselected rooftop is a gable roof, the only misselected gable rooftop in the neighbourhood, and the misselection is due to the LiDAR data being outdated: half of the building is mapped as floor level.

156 rooftops are not selected when they should have been (0.66 ha), most of them small parts (140 of them are below 100 m^2, out of the 147 misselected small rooftops) that are largely affected by the abnormal slope values. One of the largest unselected building parts (575 m^2) is misclassified because it should be split into four different parts and the divisor walls are understood as obstacles or pitched parts within it. Twelve building parts are selected when they should have not been (0.22 ha). The three largest wrongly selected parts (0.15 ha) are misselected due to outdated LiDAR data (one is the already-mentioned gable-roof building and the other two have had solar panels installed on them).

In general, the mainly misselected rooftops are small flat roofs that are highly affected by the high abnormal slope values or larger rooftops that are not correctly described by the cadastral plan or the LiDAR data. Indeed, the non-selected buildings have a higher rejected slope cells count to surface ratio $\Delta n_{\mathrm{C}}/S$, with a mean of 0.85 m^{-2}.

Finally, a total area of 0.44 ha would have been underselected, a 7% of the total selected area, but once the 100 m^2 minimum area is considered, only a total area of 0.10 ha (1.7% of the total selected area by the slope criteria, 1.9% of the area selected by the slope criteria and the 100 m^2 threshold) would have been unselected. Even if the limit were set to 50 m^2 the area left behind would have been 0.17 ha.

5.2 Expected energy savings and carbon sequestration

The total energy consumption of the L'Illa Perduda neighbourhood is estimated to be 3017 toe, being 1680 toe associated with the selected buildings' consumption, and 3.1% of it would be saved (55.3 toe) after GR installation. Following similar considerations to those taken in the previous section, if the same energy consumption per area would be extended to Valencia, out of a $E_{\mathrm{C,V}} = 217\,500\,\mathrm{toe}$ energy consumption from non-public buildings (*Anuario estadístico de Valencia 2022* 2022), 3.1% of it could be reduced, which would mean total energy savings for the city of Valencia of

$$\Delta E_{\mathrm{T,V}} = 6744\,\mathrm{toe}.$$

This would mean a net decrease in atmospheric carbon of

$$M_{\mathrm{C,T,V}} = 45\,300\,\mathrm{tCO_2} \text{ to } 49\,064\,\mathrm{tCO_2},$$

depending on the energetic mix that is used to back up the demand.

Even though if the whole population of the city were to be arranged in single-storey-buildings, the energy savings would be the highest (17% of all residential energy use), the GRs by themselves are more cost-effective when installed in buildings of three or four storeys (depending if 0.5% of dwelling energy saving is considered negligible), since they save the most amount of total energy per area, assuming the same dwelling size. Thus, probably this medium building height, which is not common in Mediterranean cities, would be the best building typology to enhance GR benefits.

5.3 Economic analysis

Choosing a concrete roofing solution is necessary to do an estimative economic analysis. The different solutions that will be considered for this are obtained from available solutions in the CYPE price generator engine (*Generador de precios de la construcción.* N.d.), choosing the default settings for each solution.

The costs of the installation of EGR based on sedum species would consist of a direct cost of 114.40 €/m^2, and a maintenance cost of 36.04 €/m^2 the first ten years. If the installation were to be done with herbaceous plants, it would have a direct cost of 105.43 €/m^2, but there is no estimate of maintenance cost (which will be assumed to be about 40 €/m^2, similar to the sedum option but a bit more expensive since herbaceous plants are a bit more difficult to maintain). The installation of a SIGR would have a direct cost of 121.34 €/m^2, similar to those of EGR, but a much higher maintenance cost, of 114.75 €/m^2.

The energy savings due to the GR installation produce an economic benefit that should be considered to finance the investment, even though it is minimal. The calculated residential energy savings per area (considering the total ground plan area of residential buildings that hold dwellings) is $\Delta E_S = 16.5\,\mathrm{kW\,h\,m^{-2}}$. Using a mean 0.2 €/kWh for the electricity price this year in Spain, the energy savings would provide a benefit of 3.3 €/m^2, a quite low result.

The carbon sequestration also should be considered, since carbon emissions are taxed in Spain. Using the price of 90 €/tCO$_2$, a direct carbon reduction per area of $7.33\,\mathrm{kgCO_2\,m^{-2}}$, and an indirect carbon reduction per area of $2.54\,\mathrm{kgCO_2\,m^{-2}}$ (if electricity is used to cover the energy demand), the associated 'benefit' would be 0.89 €/m^2. It should be kept in mind that this is not a direct benefit to the families, they do not pay for this task. It's the polluting enterprises that pay, but it is a way to translate the emissions into economic value. The economic costs overcome the benefits, but the general environmental benefits of GR installation are difficult to translate to economic revenue.

An apart discussion should be made for **single-storey and multiple-storey buildings**. For SSB, the energy savings per area are lower, $\Delta E_{\mathrm{S,SSB}} = 8.75\,\mathrm{kW\,h\,m^{-2}\,yr^{-1}}$, but for MSB, they are $\Delta E_{\mathrm{S,MSB}} = 17.3\,\mathrm{kW\,h\,m^{-2}\,yr^{-1}}$. This means that GRs save 3.46 €/m^2/yr based on energy consumption on MSB, a slightly higher result. Table 5.1 shows the financial indicators for the inversion, regarding the SSB and MSB for the three different types of roofing solutions considered. The investment is never returned, and thus, GR installation is not financially viable.

Table 5.1: Summary of the financial indicators of the investment. Total costs of investment and net present value (NPV) are given in 2023€. The costs have been calculated considering only the maintenance cost for the first 10 years, as that is the data given by the CYPE price generator. The net present value is calculated using a period of 20 years. The payback periods are marked as > 50 yr since 50 yr is the mean lifetime of GRs.

		Sedum EGR	Herbaceous EGR	SIGR
SSB	Cost of investment (€)	383.47	404.06	978.04
	Payback period (yr)	> 50	> 80	> 80
	NPV (€)	-334.22	-354.81	-928.79
MSB	Cost of investment (€)	383.47	404.06	978.04
	Payback period (yr)	> 50	> 80	> 80
	NPV (€)	-332.34	-352.93	-926.91

The project could still be economically viable if the social benefits are considered, as it can be done using the **social cost of CO_2** ($SCCO_2$). Modern studies suggest that the $SCCO_2$, i.e., the marginal cost of the impacts caused by emitting one extra tonne of carbon emissions, is higher than the usual tax values. A study from various London universities shows that by updating the PAGE-ICE model to present estimates an empirical data, the $SCCO_2$ rises to \$307 tCO_2^{-1} (in 2015 American dollars), which translates to 366.77 € tCO_2^{-1}. In the same article, when they consider the persistence of temperature-related economic impacts using an experimental distribution of this parameter the $SCCO_2$ rises drastically, with a maximum of about \$5000 tCO_2^{-1}, finding also that if the adaption rate is not high enough the cost would lower because in this scenarios the damage is so big that it does not matter to increase the CO_2 concentration. For this project to be economically viable based only on carbon emission reductions, the $SCCO_2$ should be about 3300 €/tCO_2, which is high, even though is possible based on the study. To further analyse the economic viability of the project, UHI reductions in low-rise buildings, ecosystem connectivity and societal impacts should be quantified.

5.4 AV area and GIS area differences

The AV computation yields an area of about 70% of the area calculated using GIS analysis. This decrease in the area can be explained based on five reasons.

- **Plain roofs within the selected colour regions do not account for all residential roofs.** Some of the plain roofs are painted white or grey and some residential roofs are gabled. These non-detectable rooftops account approximately for 5-8% of the residential roofs.
- **Shadows, obstacles and variation of roof colours reduce the area.** Parapet shadows account for 2-7% of the area of most roofs, obstacles are negligible, and colouring variation can affect 1-2% of the total area.
- **Parapets take some space.** Nearly all roofs have parapets delimiting them, which account for 2-3% of the rooftop surface.

- **Diagonal projection.** The pictures are not taken perfectly perpendicular to the ground level, and thus some building parts are kept hidden behind more elevated parts. This problem can hide between 10-15% of some rooftops, which could yield an error below 5%.
- **Image scaling.** As the images are not georeferenced, there is an error associated with its projection and scaling.

All these error sources combine to yield a total difference of 15-25% between the GIS calculated area and the AV delimited one. The difference would approximately be 20%, and thus a 10% error difference could be expected before a more detailed analysis is performed.

Both results are in the same order of magnitude, with a 30% difference from which a 20% can be explained by the image limitations. These results confirm that the AV can be used to estimate available areas, and can serve, if perfectioned as a tool to improve GR retrofit studies.

5.5 Error analysis

It is necessary to estimate the error of these calculations. The sources of error in this study are:

- **Cadastral geometry accuracy.** The cadastral geometry has a nominal error of 0.1 m for each point of each polygon. This error is completely negligible compared to the rest of the error sources listed here. The mean area is $86.1\,\mathrm{m}^2$ and the mean perimeter is 41.4 m. If it is assumed that the area in which the actual perimeter lies is $41.4\,\mathrm{m} \times 0.1\,\mathrm{m} = 4.2\,\mathrm{m}^2$, we could use that as an estimate of the error, which is a relative error of 4.9%. But this error would be even less. If one vertex could move relative to the others in a circle of 0.1 m, this would mean a triangular area on each side of the roof that contains said vertex, which would equate to the mean side length times the error radius. If the polygon would have n equal length sides, this area would be P/n times 0.1 m. As each vertex could move independently on the rest, this would mean an error area of $0.1\,\mathrm{m} \times P/\sqrt{n}$. If $n = 4$ is assumed for the mean roof, this would be half the error that has been noted before, i.e. 2.5%.
- **Threshold selection.** The selection of the minimum area in $100\,\mathrm{m}^2$ is arbitrary. Even though it is based on other results and concrete reasoning related to the slope criterium, it is still quite arbitrary. Thus, an error should be assigned to it. When thinking about which threshold to choose, an area of $50\,\mathrm{m}^2$ was also discussed as a minimum working area that could easily be put together on most rooftops. When using $50\,\mathrm{m}^2$ as a threshold, the available area grows from 4.83 ha to 5.15 ha, an increase of 6.6%. This proportion could be linked to the induced error when selecting a threshold.
- **Height measurements and slope calculations.** The LiDAR data is imported into the calculations, it is treated as a layer comprised of $1\,\mathrm{m}^2$ cells. This discretization introduces the abnormal values in the slope analysis, since when the edge of a building lies within one of the cells, the height value will be between the height of the roof and the ground level. To tackle this problem, ignoring the cells whose value is higher than 30° produces the elimination of 43% of the data. This in the end is not as problematic, since it yields a total error of 1.9% (see section 5.1, pitched roofs and terraces selection goodness), but in a different analysis, it could be about 8.6% (regarding the total misselected area).

- **Obstacles in the selected roofs.** Based on the aerial imaging results, the unexplained difference between the GIS result and the AV result is about 10%, which, being conservative, could be used as the error to estimate the obstacle presence on rooftops.

If these error sources are assumed to be independent, combining the different errors would yield a total error of 10-15%. An assumable error for a first estimation of the potential.

Chapter 6

Conclussions

This study has succeeded in achieving its main objectives. Using GIS and AV algorithms, a methodology has been developed to select and characterise suitable rooftops within the city of Valencia, using L'Illa Perduda as a case study. Once the buildings are selected, the available areas have been calculated and the potential benefits have been evaluated using statistical experimental data. In the end, a viability study has been performed for the GR installation in L'Illa Perduda.

To **select the suitable buildings**, geographical information systems have been used. From public cadastral information, the different building parts have been obtained, while the LiDAR covering of the city has yielded the height data necessary to study the building height and slope. Out of this data, a minimum threshold in rooftop area of $100\,\mathrm{m}^2$ and a maximum value for roof pitch of 10° has been applied. Also, to avoid the inclusion of small one-storey parts that are used as terraces or courtyards, a minimum surface of $500\,\mathrm{m}^2$ has been imposed for SSB, as a conservative measure. After the application of these requirements, a total area of 4.83 ha (5.33 ha without the SSB limit) has been selected as an available area for greening, 21% of the total neighbourhood area (23%), and 48% of the total rooftop area (54%). Also, 1.86 ha, 39% of the selected area is occupied by roofs less than 25 m high, which would improve the connectivity of green areas in the city, and 0.56 ha (10% of the selected area) corresponds to rooftops under 10 m high, which could help reduce the UHI effect.

The two main **requirements** to select a rooftop as greenable have been studied to avoid arbitrariness in their definition or calculation. The area limit has been shown to approximate the separation between two distributions of roofs relating to their calculated slope and their geometrical parameters. The slope criterium has been verified, as it induces little error in pitched or obstacle-filled roof detecting, and the parameters used to apply it have been analysed. Both requirements work well together, even though further analysis with a bigger statistic is needed to confirm their goodness.

The **AV algorithms** have been successfully applied to detect flat roofs in the neighbourhood using high-quality aerial images. The semi-automatic identification yields an area of 3.05 ha out of flat roofs, 71% of the residential roofs selected by the GIS study. A 20% of the 30% difference can be explained by the limitations of the presented method and, while a more detailed study should be performed, only a 10% error is sufficient proof to claim that the AV techniques can be satisfactorily applied to calculate the potential for GR retrofit.

Once the roof selection is finished, using experimental data, the study has resulted in an **expected carbon sequestration** of $354.2\,\mathrm{tCO_2\,yr^{-1}}$ to $484.1\,\mathrm{tCO_2\,yr^{-1}}$, depending on the energetic mix that is used to back the energetic demand. Also, the study calculated **energy savings** of $55.3\,\mathrm{toe\,yr^{-1}}$ for residential buildings, 3% of the energy consumption in these buildings, and 1% of the energy consumption of the neighbourhood. If some estimations are performed to also extend the results to non-residential buildings, a total energy savings of $65.5\,\mathrm{toe\,yr^{-1}}$ would be expected, 4% of the energy consumed in the greened buildings. Green roofs, even though they are most efficient in SSB, in the sense that a higher percentage of the total energy is saved, help save the most energy when installed in MSB with four roofs or more.

The project of installing green roofs in urban areas is not **financially viable**, as expected. The derivate benefits of energy savings only rise to $3.3\,€\,\mathrm{yr^{-1}}$, which is not comparable with annual costs of about $40\,€\,\mathrm{yr^{-1}}$. Nonetheless, the project could be **economically feasible**, if the $SCCO_2$ were to be considered, as well as the UHI reductions, the ecosystem benefits and the societal impacts, but a more refined study is necessary to affirm the project's economic viability.

Personally, this project has served as an introduction to research in sustainability-related fields and has helped me to work closer to a problem I am really invested in tackling. Moreover, it has granted me valuable experience and has taught me that things are usually more complex than they seem.

Part II

Budget

Chapter 7

Project budget

This Chapter will produce a detailed budget for the finalisation of the study to calculate the potential for green roof installation in the L'Illa Perduda neighbourhood.

7.1 Project time frame and budget disaggregation

The analysis was performed over about three months, considering the time to carry out the literature review, the data acquisition, the artificial vision algorithm development, the design of the methodology, the calculations and both the energetic and economic analysis, and the writing of this report. Such a time frame was possible thanks to the funding and guidance of the Instituto de Ingeniería Energética de Valencia, as a result of my internship there.

The budget here performed will disaggregate total costs into human resources cost, equipment amortisation and general costs, over three months.

Human resources

For the completion of this study, the human resources needed were a junior engineer, an assistant professor, and a permanent professor. The tasks that the junior engineer performed were:

- **Literature review.** Literature research in NBS, green roofing solutions, green roof benefits, green roof retrofit analysis, artificial vision, GIS analysis and benefits estimation.
- **Data gathering.** Data downloading from different public repositories, and image taking with Google Earth.
- **Algorithm development.** Production of a functioning algorithm to calculate available area from aerial imagery via artificial vision methods with MATLAB.
- **Methodology design.** Development of a methodology to classify the different buildings, setting adequate criteria and indicators, and identifying different roofing solutions throughout multiple images.

- **Calculations and analysis.** Classify the buildings, calculate available areas and derivate energetic and economic analysis.
- **Report writing.** Drafting and producing the present report, as well as most figures within it.

The assistant professor and the permanent professor carried out the guiding of the structure and development of this project and revising the findings and work of the junior engineer. They have controlled the completion of the study by email communication, in-person and online meetings and delicate revision of the present document. Table 7.1 summarises the human resources budget.

Table 7.1: Human resources cost summary.

Task	Time (h)	Position	Unit cost (€ h^{-1})	Cost (EURO)
Literature Review	60	Junior engineer	30	1800
Data gathering	20	Junior engineer	30	600
Algorithm development	70	Junior engineer	30	2100
Methodology design	50	Junior engineer	30	1500
Calculation and analysis	40	Junior engineer	30	1200
Report writing	60	Junior engineer	30	1800
Guidance and revision	20	Assistant professor	60	1200
Guidance and revision	20	Permanent professor	80	1600
Total	340			11800

Software and hardware amortisation costs

The equipment used to develop this project consists of three computers, belonging to the junior engineer, the assistant professor and the permanent professor. The software used for the completion of the project is in its majority, free software: GIS analysis (QGIS), image processing (FIJI), and typesetting (VSCode). Also, Google Earth Pro, used to gather images, is of free use. MATLAB and Microsoft 365 are the only software that is not open-source nor free to use, but it has been used under the licensee agreement with the UPV, so it is of free use for its community. Table 7.2 shows a summary of the equipment budget.

Table 7.2: Hardware amortisation summary.

Equipment	Price (€)	Full amortisation	Use	Cost (€)
ASUS Rog (Junior engineer)	1000	5 yr	3 months	50
PC intel i7 (Assistant professor)	800	5 yr	Half a month	6.7
MSI Prestige (Permanent professor)	1000	5 yr	Half a month	8.3
Total				65

General costs

The general costs here considered consist of travel expenses, internet connection and electricity bills. At the moment, given that in the Comunidad Valenciana public transportation is free for people under thirty, the travel expenses are none. Table 7.3 shows the summary of the general costs.

Table 7.3: General costs summary.

Expense	Quantity	Unitary cost	Cost (€)
Travel	50 trips	0 €/trip	0
Electricity	3 months	40 €/month	120
Internet	3 months	20 €/month	80
Total			200

7.2 Budget summary

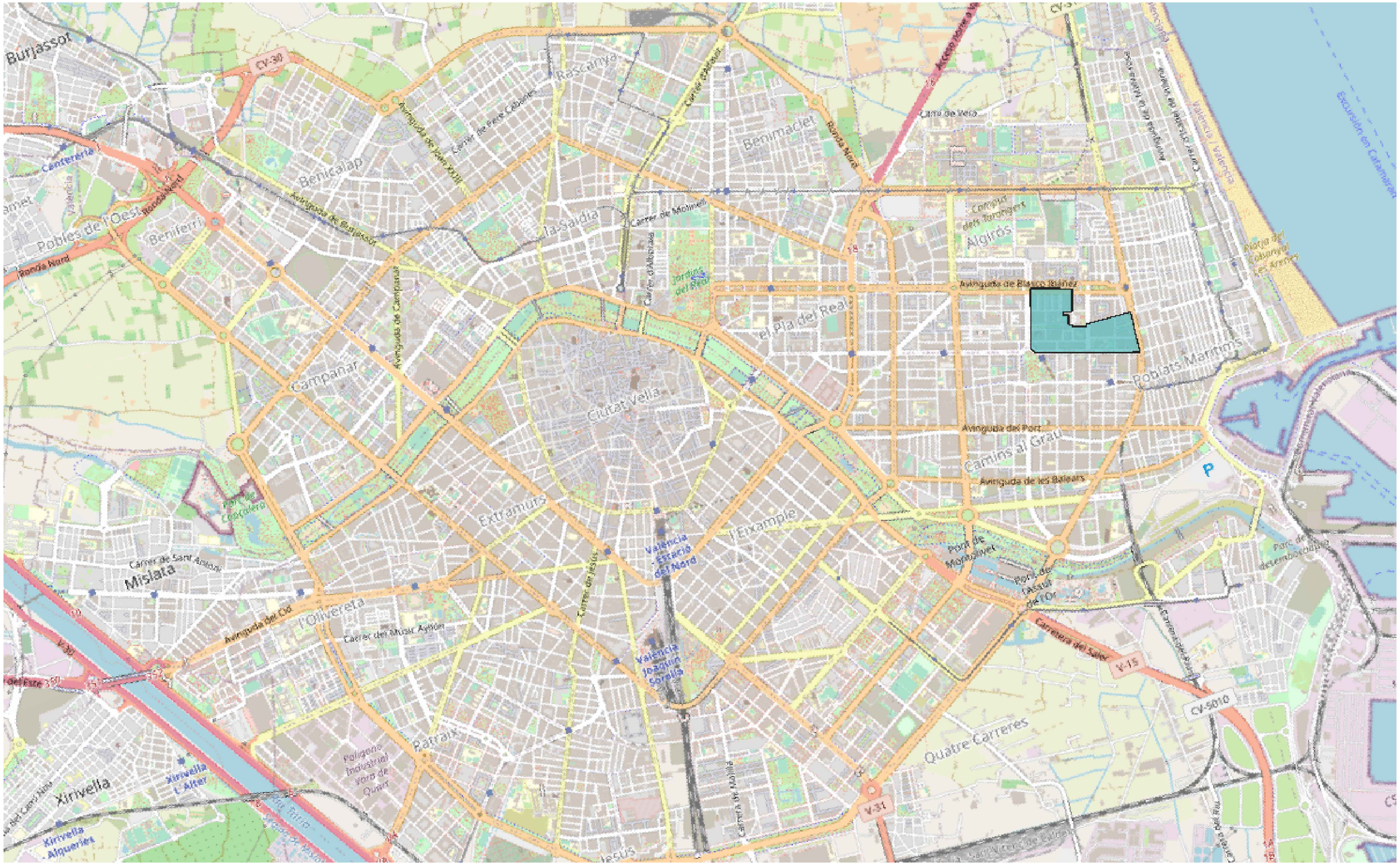
Now, the project costs are summarised into an aggregated gross budget, also considering a profit margin of 6% and the corresponding taxes. As table 7.4 shows, the total cost of the project is fifteen thousand four hundred eighty-two euros with forty-nine cents.

Table 7.4: Contractual budget detail.

Expense	Cost (€)
Human resources	11800
Equipment amortisation	65
General costs	200
Gross execution budget	12065
Industrial profit (6%)	723.9
Industrial budget	12788.9
Consumption tax (IVA 21%)	2685.67
Contractual budget	15474.57

Part III

Blueprints



TRABAJO FINAL DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

ESCUELA TÉCNICA SUPERIOR INGENIERÍA INDUSTRIAL VALENCIA

Proyecto: Calculation of the decarbonisation potential of the installation of green roofs in the L'Illa Perduda neighbourhood in València

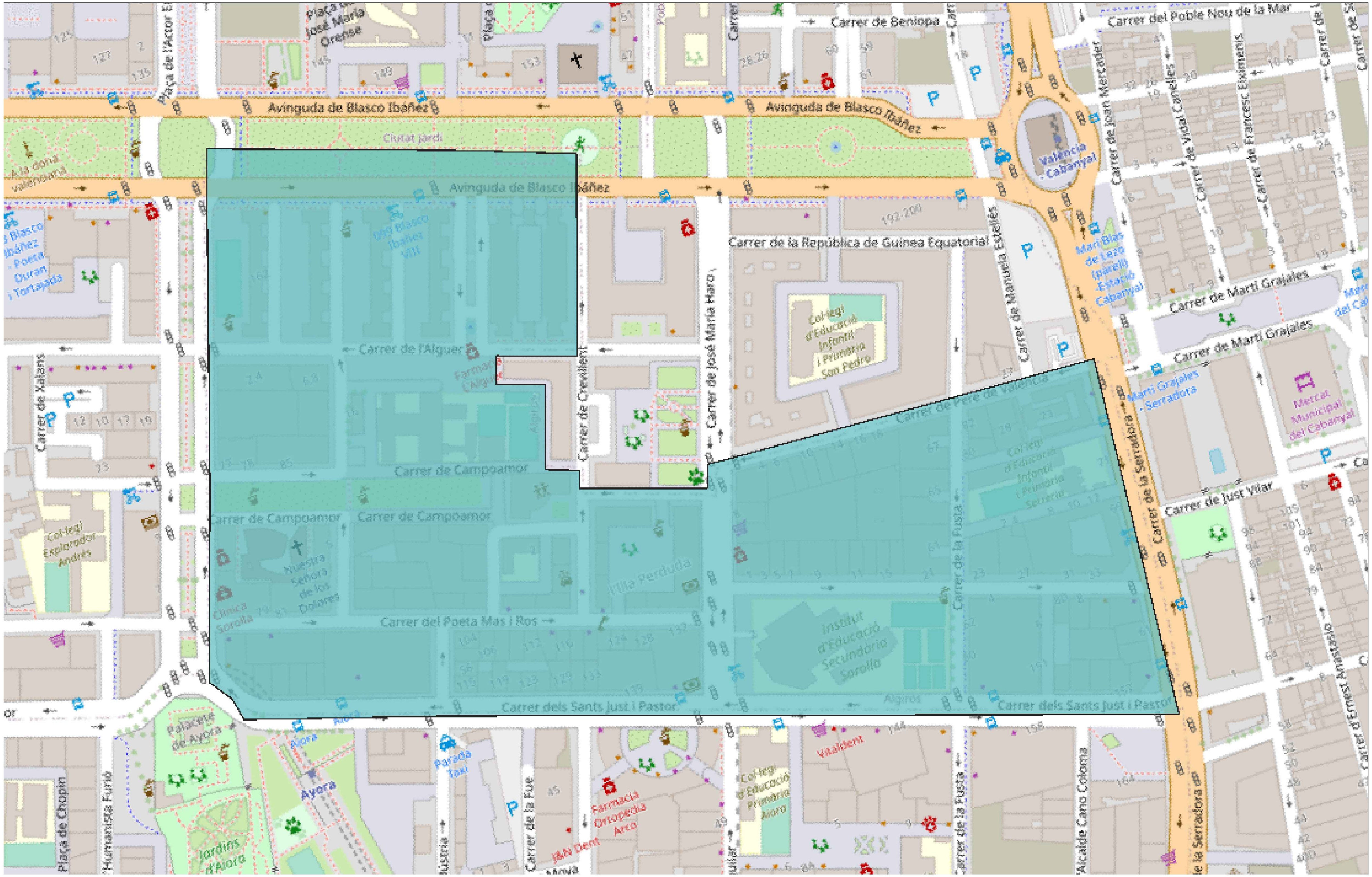
Plano: Site of the L'Illa Perduda neighbourhood

Autor: Max Zayas Orihuela

Fecha: Septiembre 2023

Escala: 1:32000

Nº Plano: 1



TRABAJO FINAL DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

ESCUELA TÉCNICA SUPERIOR INGENIERÍA INDUSTRIAL VALENCIA

Proyecto: Calculation of the decarbonisation potential of the installation of green roofs in the L'Illa Perduda neighbourhood in València

Plano: Site of the L'Illa Perduda neighbourhood

Autor: Max Zayas Orihuela

Fecha: Septiembre 2023

Escala: 1:3600

Nº Plano: 2

Part IV

Annexes

Appendix A

Aerial images of L'Illa Perduda

Figure A.1: Resulting mask of the first aerial image of the L'Illa Perduda neighbourhood. The original image is shown, and the resulting mask is overlapped onto it in a blueish colour. The base image is the cropped result of the GEP LLC 2023 picture (2020) and the overlapped mask is produced using MATLAB MathWorks 2022.

Figure A.2: Resulting mask of the second aerial image of the L'Illa Perduda neighbourhood. The original image is shown, and the resulting mask is overlapped onto it in a blueish colour. The base image is the cropped result of the GEP LLC 2023 picture (2020) and the overlapped mask is produced using Matlab MathWorks 2022.

Figure A.3: Resulting mask of the third aerial image of the L'Illa Perduda neighbourhood. The original image is shown, and the resulting mask is overlapped onto it in a blueish colour. The base image is the cropped result of the GEP LLC 2023 picture (2020) and the overlapped mask is produced using Matlab MathWorks 2022.

Figure A.4: Resulting mask of the fourth aerial image of the L'Illa Perduda neighbourhood. The original image is shown, and the resulting mask is overlapped onto it in a blueish colour. The base image is the cropped result of the GEP LLC 2023 picture (2020) and the overlapped mask is produced using MATLAB MathWorks 2022.

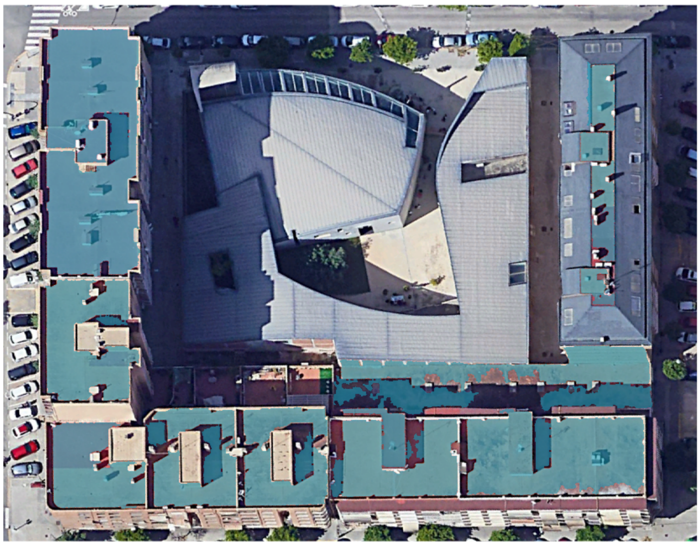


Figure A.5: Resulting mask of the fifth aerial image of the L'Illa Perduda neighbourhood. The original image is shown, and the resulting mask is overlapped onto it in a blueish colour. The base image is the cropped result of the GEP LLC 2023 picture (2020) and the overlapped mask is produced using MATLAB MathWorks 2022.

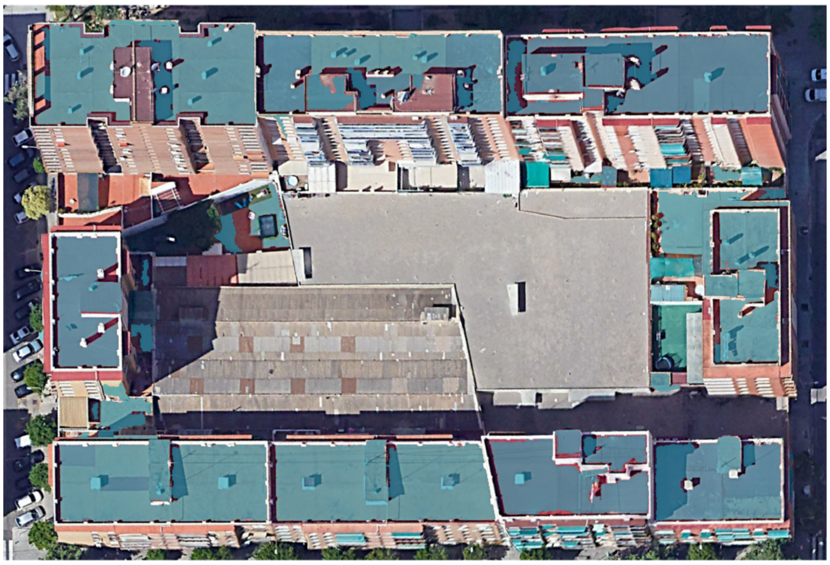


Figure A.6: Resulting mask of the sixth aerial image of the L'Illa Perduda neighbourhood. The original image is shown, and the resulting mask is overlapped onto it in a blueish colour. The base image is the cropped result of the GEP LLC 2023 picture (2020) and the overlapped mask is produced using MATLAB MathWorks 2022.

Figure A.7: Resulting mask of the seventh aerial image of the L'Illa Perduda neighbourhood. The original image is shown, and the resulting mask is overlapped onto it in a blueish colour. The base image is the cropped result of the GEP LLC 2023 picture (2020) and the overlapped mask is produced using MATLAB MathWorks 2022.

Figure A.8: Resulting mask of the eighth aerial image of the L'Illa Perduda neighbourhood. The original image is shown, and the resulting mask is overlapped onto it in a blueish colour. The base image is the cropped result of the GEP LLC 2023 picture (2020) and the overlapped mask is produced using MATLAB MathWorks 2022.

Figure A.9: Resulting mask of the ninth aerial image of the L'Illa Perduda neighbourhood. The original image is shown, and the resulting mask is overlapped onto it in a blueish colour. The base image is the cropped result of the GEP LLC 2023 picture (2020) and the overlapped mask is produced using MATLAB MathWorks 2022.

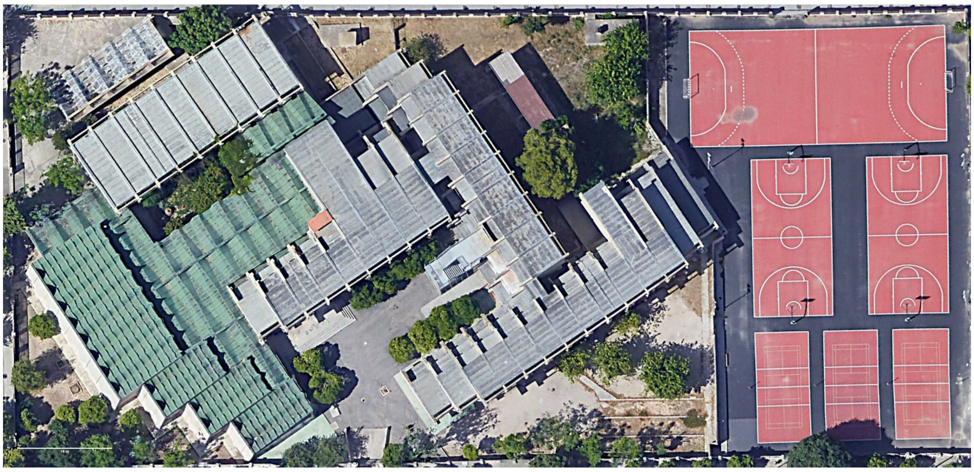


Figure A.10: Resulting mask of the tenth aerial image of the L'Illa Perduda neighbourhood. Since there are no flat-roof buildings, there is no mask produced. The original image is shown, and the resulting mask is overlapped onto it in a blueish colour. The base image is the cropped result of the GEP LLC 2023 picture (2020) and the overlapped mask is produced using Matlab MathWorks 2022.

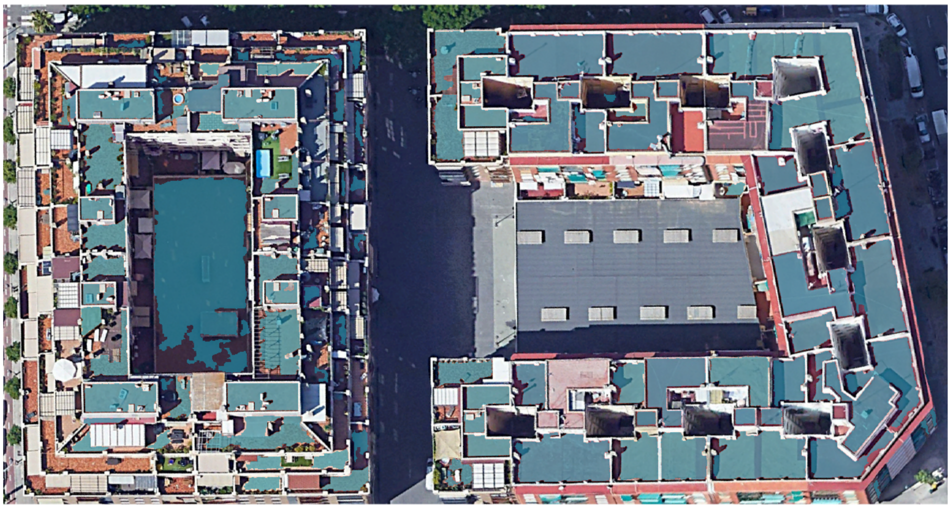


Figure A.11: Resulting mask of the eleventh aerial image of the L'Illa Perduda neighbourhood. The original image is shown, and the resulting mask is overlapped onto it in a blueish colour. The base image is the cropped result of the GEP LLC 2023 picture (2020) and the overlapped mask is produced using Matlab MathWorks 2022.

Figure A.12: Resulting mask of the twelfth aerial image of the L'Illa Perduda neighbourhood. The original image is shown, and the resulting mask is overlapped onto it in a blueish colour. The base image is the cropped result of the GEP LLC 2023 picture (2020) and the overlapped mask is produced using MATLAB MathWorks 2022.

Figure A.13: Resulting mask of the thirteenth aerial image of the L'Illa Perduda neighbourhood. The original image is shown, and the resulting mask is overlapped onto it in a blueish colour. The base image is the cropped result of the GEP LLC 2023 picture (2020) and the overlapped mask is produced using MATLAB MathWorks 2022.

Appendix B

MATLAB code for AV area detection

```
%% ==========================================================
%
% Script for AV area detection for green roof retrofit
% Author: Max Zayas Orihuela
% Institution: Instituto de IngenierÃ a EnergÃ©tica (UPV)
%
% ==========================================================
%
% DISCLAIMER: This code has not been cleaned or polished. It is not
    meant
% to be used by people outside the institution.
%
% ==========================================================

%% roofType struct loading

% Roof type structure loading
load("MATLABData\roofType.mat","roofType");

%% Image stuct creation / import

tAerialImage = Tiff(['C:\Users\maxzy\Desktop\Academia\Research\UPV\
   Catenerg\Trabajo\MALTAB\Fotos\Fotos para MATLAB\Tiff Images' ...
    '\LIlla_Perduda_1.tif']);

AerialImage = tAerialImage.read;
AerialImageHSV = rgb2hsv(AerialImage);

[~,imageName,~] = fileparts(tAerialImage.FileName);
fileStructPath = "MATLABData\imageMasks\" + imageName + ".mat";
```

```
if isfile(fileStructPath)
    load(fileStructPath);
    imageStruct.image.tiff = tAerialImage;
else
    imageStruct = struct();
    imageStruct.name = imageName;
    imageStruct.image = struct();
    imageStruct.image.tiff = tAerialImage;
    imageStruct.image.RGB = AerialImage;
    imageStruct.image.HSV = AerialImageHSV;
    len = length(roofType);
    for i = 1:len
        imageStruct.roofTypes(i) = struct("name",{roofType(i).name
           },"colorRegion",{struct("name",{roofType(i).colorRegions
           .name})});
    end
    imageStruct.image.medianFiltered = struct();
    imageStruct.image.medianFiltered.RGB = cat(3,medfilt2(
       AerialImage(:,:,1),[5,5]),medfilt2(AerialImage(:,:,2),[5,5])
       ,medfilt2(AerialImage(:,:,3),[5,5]));
    imageStruct.image.medianFiltered.HSV = rgb2hsv(imageStruct.
       image.medianFiltered.RGB);
end

%% Pixel detection

answer = questdlg("Would you like to add a new roofType for this
   image?");
while answer=="Yes"

    roofName = input("Please, write down the name of the new
       roofType.\n","s");
    colorRegionName = input("Please, write down the name of the new
        colorRegion for this roofType.\n","s");
    pixelClicks = selectRoofPixels(AerialImage);

    newRoofType = struct();
    newRoofType.name = convertCharsToStrings(roofName);

    colorRegion = struct();
    colorRegion.name = convertCharsToStrings(colorRegionName);
    colorRegion.pixelValuesRGB = getPixelValues(AerialImage,
       pixelClicks);
```

```
    colorRegion.pixelValuesHSV = getPixelValues(AerialImageHSV,
       pixelClicks);
    colorRegion.statistics = getStatistics(colorRegion);
    newRoofType.colorRegions = [];
    newRoofType.colorRegions = [newRoofType.colorRegions
       colorRegion];

    roofType = [roofType newRoofType];

    answer = questdlg("Would you like to add a new roofType for
       this image?");
end

answer = questdlg("Would you like to add a new colorRegion for an
   existing roofType?");

while answer=="Yes"

    [idx,tf] = listdlg('ListString',[roofType.name],'SelectionMode'
       ,'single','PromptString',"Select the roofType to modify:");

    if isempty(idx)
        break
    end

    colorRegionName = input("Please, write down the name of the new
        colorRegion for this roofType.\n","s");
    pixelClicks = selectRoofPixels(AerialImage);

    newColorRegion = struct();
    newColorRegion.name = convertCharsToStrings(colorRegionName);
    newColorRegion.pixelValuesRGB = getPixelValues(AerialImage,
       pixelClicks);
    newColorRegion.pixelValuesHSV = getPixelValues(AerialImageHSV,
       pixelClicks);

    newColorRegion.statistics = getStatistics(newColorRegion);

    roofType(idx).colorRegions = [roofType(idx).colorRegions
       newColorRegion];

    answer = questdlg("Would you like to add a new colorRegion for
       an existing roofType?");
end
```

```
answer = questdlg("Would you like to add more points to an existing
    colorRegion?");

while answer=="Yes"

    [roofTypeIdx,tf] = listdlg('ListString',[roofType.name],'
       SelectionMode','single','PromptString',"Select the roofType
       to modify:");

    if isempty(roofTypeIdx)
         break
    end

    [colorRegionIdx,tf] = listdlg('ListString',[roofType(
       roofTypeIdx).colorRegions.name],'SelectionMode','single','
       PromptString',"Select the colorRegion to modify:");

    if isempty(colorRegionIdx)
         break
    end

    pixelClicks = selectRoofPixels(AerialImage);

    roofType(roofTypeIdx).colorRegions(colorRegionIdx).
       pixelValuesRGB = [roofType(roofTypeIdx).colorRegions(
       colorRegionIdx).pixelValuesRGB; getPixelValues(AerialImage,
       pixelClicks)];
    roofType(roofTypeIdx).colorRegions(colorRegionIdx).
       pixelValuesHSV = [roofType(roofTypeIdx).colorRegions(
       colorRegionIdx).pixelValuesHSV; getPixelValues(
       AerialImageHSV,pixelClicks)];
    roofType(roofTypeIdx).colorRegions(colorRegionIdx).statistics =
        getStatistics(roofType(roofTypeIdx).colorRegions(
       colorRegionIdx));

    answer = questdlg("Would you like to add more points to an
       existing colorRegion in this image?");
end

%% Save the roofType struct

save("MATLABData\roofType","roofType");

%% Create mask for roofType
```

```
answer = questdlg("Would you like to create a mask for this image for some roofType?");

while answer=="Yes"

    medianFiltered = 0;

    [idx,tf] = listdlg('ListString',[roofType.name],'SelectionMode','single','PromptString',"Select the roofType to create the mask for:");

    if isempty(idx)
        break
    end

    imageStruct = createMask(imageStruct,roofType,idx,medianFiltered);
    %imageStruct = filterMask(imageStruct,idx);

    answer = questdlg("Would you like to create another mask for this image for some roofType?");
end

answer = questdlg("Would you like to create a mask for this median filtered image for some roofType?");

while answer=="Yes"

    medianFiltered = 1;

    [idx,tf] = listdlg('ListString',[roofType.name],'SelectionMode','single','PromptString',"Select the roofType to create the mask for:");

    if isempty(idx)
        break
    end

    imageStruct = createMask(imageStruct,roofType,idx,medianFiltered);
    % imageStruct = filterMask(imageStruct,idx);

    answer = questdlg("Would you like to create another mask for this median filtered image for some roofType?");
end
```

```
%% Trials I. Filter grey mask

    closedMask = imclose(imageStruct.roofTypes(2).colorRegion(1).
       meanStdMask.total,[0 1 0; 1 1 1; 0 1 0]);
    filledMask = myFill(closedMask,800);
    openedFilledMask = myOpen(filledMask,[0 1 0 ; 1 1 1 ; 0 1
       0],1,1);

    % Suppress smaller blobs
    % Extract blobs and blob area
    imLabels = bwlabel(openedFilledMask);
    props = regionprops(openedFilledMask,imLabels,'Area');

    % Surpress small blobs
    k = find([props.Area] < 300);
    suppressedLabels = ismember(imLabels,k);
    filteredMask = xor(openedFilledMask,suppressedLabels);
    figure("Name","First filtered mask");
    imshow(filteredMask);

    closedMask = myClose(filteredMask,[0 1 0 ; 1 1 1 ; 0 1 0],2,2);
    filledFilteredMask = myFill(closedMask,700);

    % Suppress smaller blobs again
    % Extract blobs and blob area
    imLabels = bwlabel(filledFilteredMask);
    props = regionprops(filledFilteredMask,imLabels,'Area');

    % Surpress small blobs
    k = find([props.Area] < 1000);
    suppressedLabels = ismember(imLabels,k);
    filteredMask = xor(filledFilteredMask,suppressedLabels);
    figure("Name","Twice filtered mask");
    imshow(filteredMask);

    % Finally close the image
    closedMask = myClose(filteredMask,[0 1 0 ; 1 1 1 ; 0 1 0],6,6);
    filledFilteredMask = myFill(closedMask,1000);
    figure("Name","Final filled closed filtered mask");
    imshow(filledFilteredMask);


 %% Trials I. Filter grey mask 2
```

```
closedMask = imclose(imageStruct.roofTypes(2).colorRegion(2).
   minMaxMask.total,[0 1 0; 1 1 1; 0 1 0]);
filledMask = myFill(closedMask,800);
openedFilledMask = myOpen(filledMask,[0 1 0 ; 1 1 1 ; 0 1
   0],1,1);

% Suppress smaller blobs
% Extract blobs and blob area
imLabels = bwlabel(openedFilledMask);
props = regionprops(openedFilledMask,imLabels,'Area');

% Surpress small blobs
k = find([props.Area] < 300);
suppressedLabels = ismember(imLabels,k);
filteredMask = xor(openedFilledMask,suppressedLabels);
figure("Name","First filtered mask");
imshow(filteredMask);

closedMask = myClose(filteredMask,[0 1 0 ; 1 1 1 ; 0 1 0],2,2);
filledFilteredMask = myFill(closedMask,700);

% Suppress smaller blobs again
% Extract blobs and blob area
imLabels = bwlabel(filledFilteredMask);
props = regionprops(filledFilteredMask,imLabels,'Area');

% Surpress small blobs
k = find([props.Area] < 1000);
suppressedLabels = ismember(imLabels,k);
filteredMask = xor(filledFilteredMask,suppressedLabels);
figure("Name","Twice filtered mask");
imshow(filteredMask);

% Finally close the image
closedMask = myClose(filteredMask,[0 1 0 ; 1 1 1 ; 0 1 0],6,6);
filledFilteredMask = myFill(closedMask,1000);
figure("Name","Final filled closed filtered mask");
imshow(filledFilteredMask);


%%
imageStruct.roofTypes(2).colorRegion(2).filteredMask.MinMaxMask
    = filledFilteredMask;
visualizeMask(imageStruct.image.RGB,filledFilteredMask);
```

```
%% Trials I. Filter individual masks. Grey (Light)

% Grey (Light)
imageStruct.roofTypes(2).colorRegion(1).filteredMask = struct();

imageStruct.roofTypes(2).colorRegion(1).filteredMask.MeanStdMask =
   filterIndividualGreyMask(imageStruct.roofTypes(2).colorRegion(1)
   .meanStdMask.total,600,1200);
imageStruct.roofTypes(2).colorRegion(1).filteredMask.MinMaxMask =
   filterIndividualGreyMask(imageStruct.roofTypes(2).colorRegion(1)
   .minMaxMask.total,600,1200);

visualizeMask(imageStruct.image.RGB,imageStruct.roofTypes(2).
   colorRegion(1).filteredMask.MinMaxMask);

% Blob OR
imageStruct.roofTypes(2).colorRegion(1).filteredMask.total = struct
   ();
imageStruct.roofTypes(2).colorRegion(1).filteredMask.total.minMax =
    struct();
imageStruct.roofTypes(2).colorRegion(1).filteredMask.total.minMax.
   blobOR = blobOR(imageStruct.roofTypes(2).colorRegion(1).
   filteredMask.RGB.filteredMinMaxMask,imageStruct.roofTypes(2).
   colorRegion(1).filteredMask.HSV.filteredMinMaxMask);
imageStruct.roofTypes(2).colorRegion(1).filteredMask.total.minMax.
   OR = imageStruct.roofTypes(2).colorRegion(1).filteredMask.RGB.
   filteredMinMaxMask|imageStruct.roofTypes(2).colorRegion(1).
   filteredMask.HSV.filteredMinMaxMask;
imageStruct.roofTypes(2).colorRegion(1).filteredMask.total.meanStd
   = struct();
imageStruct.roofTypes(2).colorRegion(1).filteredMask.total.meanStd.
   blobOR = blobOR(imageStruct.roofTypes(2).colorRegion(1).
   filteredMask.RGB.filteredMeanStdMask,imageStruct.roofTypes(2).
   colorRegion(1).filteredMask.HSV.filteredMeanStdMask);
imageStruct.roofTypes(2).colorRegion(1).filteredMask.total.meanStd.
   OR = imageStruct.roofTypes(2).colorRegion(1).filteredMask.RGB.
   filteredMeanStdMask|imageStruct.roofTypes(2).colorRegion(1).
   filteredMask.HSV.filteredMeanStdMask;

% AND
imageStruct.roofTypes(2).colorRegion(1).filteredMask.total.AND =
   and(and(imageStruct.roofTypes(2).colorRegion(1).filteredMask.HSV
   .filteredMinMaxMask,imageStruct.roofTypes(2).colorRegion(1).
   filteredMask.HSV.filteredMeanStdMask), ...
```

```
and(imageStruct.roofTypes(2).colorRegion(1).filteredMask.RGB.
   filteredMinMaxMask,imageStruct.roofTypes(2).colorRegion(1).
   filteredMask.RGB.filteredMeanStdMask));

% Fuse Light
imageStruct.roofTypes(2).mask.fusedFilteredMask.light.total.blobOR
   = blobOR(blobOR(imageStruct.roofTypes(2).colorRegion(1).
   filteredMask.total.minMax.blobOR,imageStruct.roofTypes(2).
   colorRegion(1).filteredMask.total.minMax.OR),blobOR(imageStruct.
   roofTypes(2).colorRegion(1).filteredMask.total.meanStd.blobOR,
   imageStruct.roofTypes(2).colorRegion(1).filteredMask.total.
   meanStd.OR));

%% Trials I. Crop blobs
croppedMask = cropBlobs(imageStruct.roofTypes(2).colorRegion(2).
   filteredMask.MinMaxMask);

%% Visualize it
visualizeMask(imageStruct.image.RGB,croppedMask);

%% Finally close the mask

% First, fuse super close blobs.
imageStruct.roofTypes(2).mask.finalMask = myFill(myClose(
   croppedMask,ones(3),3,3),200);
visualizeMask(imageStruct.image.RGB,imageStruct.roofTypes(2).mask.
   finalMask);

% Second, fuse individual blobs.
lastLabels = bwlabel(imageStruct.roofTypes(2).mask.finalMask);
blobs = unique(lastLabels(:));

finalBlobs = zeros(size(lastLabels));
for i = blobs(blobs>0)'
    blobMask = ismember(lastLabels,i);
    blobClosedMask = myClose(blobMask,ones(3),10,10);
    finalBlobs = finalBlobs|blobClosedMask;
end

imageStruct.roofTypes(2).mask.finalMask = myFill(finalBlobs,300);
visualizeMask(imageStruct.image.RGB,imageStruct.roofTypes(2).mask.
   finalMask);

%% Trials I. Filter individual masks. Red (Light)
```

```
% Red (Light)
imageStruct.roofTypes(1).colorRegion(1).filteredMask = struct();

imageStruct.roofTypes(1).colorRegion(1).filteredMask.RGB = struct()
   ;
imageStruct.roofTypes(1).colorRegion(1).filteredMask.RGB.
   filteredMeanStdMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(1).meanStdMask.RGB,400,1200);
imageStruct.roofTypes(1).colorRegion(1).filteredMask.RGB.
   filteredMinMaxMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(1).minMaxMask.RGB,400,1200);

imageStruct.roofTypes(1).colorRegion(1).filteredMask.HSV = struct()
   ;
imageStruct.roofTypes(1).colorRegion(1).filteredMask.HSV.
   filteredMeanStdMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(1).meanStdMask.HSV,400,1200);
imageStruct.roofTypes(1).colorRegion(1).filteredMask.HSV.
   filteredMinMaxMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(1).minMaxMask.HSV,400,1200);

% Blob OR
imageStruct.roofTypes(1).colorRegion(1).filteredMask.total = struct
   ();
imageStruct.roofTypes(1).colorRegion(1).filteredMask.total.minMax =
    struct();
imageStruct.roofTypes(1).colorRegion(1).filteredMask.total.minMax.
   blobOR = blobOR(imageStruct.roofTypes(1).colorRegion(1).
   filteredMask.RGB.filteredMinMaxMask,imageStruct.roofTypes(1).
   colorRegion(1).filteredMask.HSV.filteredMinMaxMask);
imageStruct.roofTypes(1).colorRegion(1).filteredMask.total.minMax.
   OR = imageStruct.roofTypes(1).colorRegion(1).filteredMask.RGB.
   filteredMinMaxMask|imageStruct.roofTypes(1).colorRegion(1).
   filteredMask.HSV.filteredMinMaxMask;
imageStruct.roofTypes(1).colorRegion(1).filteredMask.total.meanStd
   = struct();
imageStruct.roofTypes(1).colorRegion(1).filteredMask.total.meanStd.
   blobOR = blobOR(imageStruct.roofTypes(1).colorRegion(1).
   filteredMask.RGB.filteredMeanStdMask,imageStruct.roofTypes(1).
   colorRegion(1).filteredMask.HSV.filteredMeanStdMask);
imageStruct.roofTypes(1).colorRegion(1).filteredMask.total.meanStd.
   OR = imageStruct.roofTypes(1).colorRegion(1).filteredMask.RGB.
   filteredMeanStdMask|imageStruct.roofTypes(1).colorRegion(1).
   filteredMask.HSV.filteredMeanStdMask;
```

```
%% Trials I. Filter individual masks. Orange (Light)
imageStruct.roofTypes(1).colorRegion(4).filteredMask = struct();

imageStruct.roofTypes(1).colorRegion(4).filteredMask.RGB = struct()
   ;
imageStruct.roofTypes(1).colorRegion(4).filteredMask.RGB.
   filteredMeanStdMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(4).meanStdMask.RGB,400,1200);
imageStruct.roofTypes(1).colorRegion(4).filteredMask.RGB.
   filteredMinMaxMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(4).meanStdMask.RGB,400,1200);

imageStruct.roofTypes(1).colorRegion(4).filteredMask.HSV = struct()
   ;
imageStruct.roofTyqpes(1).colorRegion(4).filteredMask.HSV.
   filteredMeanStdMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(4).meanStdMask.HSV,400,1200);
imageStruct.roofTypes(1).colorRegion(4).filteredMask.HSV.
   filteredMinMaxMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(4).minMaxMask.HSV,400,1200);

% Blob OR
imageStruct.roofTypes(1).colorRegion(4).filteredMask.total = struct
   ();
imageStruct.roofTypes(1).colorRegion(4).filteredMask.total.minMax =
    struct();
imageStruct.roofTypes(1).colorRegion(4).filteredMask.total.minMax.
   blobOR = blobOR(imageStruct.roofTypes(1).colorRegion(4).
   filteredMask.RGB.filteredMinMaxMask,imageStruct.roofTypes(1).
   colorRegion(4).filteredMask.HSV.filteredMinMaxMask);
imageStruct.roofTypes(1).colorRegion(4).filteredMask.total.minMax.
   OR = imageStruct.roofTypes(1).colorRegion(4).filteredMask.RGB.
   filteredMinMaxMask|imageStruct.roofTypes(1).colorRegion(4).
   filteredMask.HSV.filteredMinMaxMask;
imageStruct.roofTypes(1).colorRegion(4).filteredMask.total.meanStd
   = struct();
imageStruct.roofTypes(1).colorRegion(4).filteredMask.total.meanStd.
   blobOR = blobOR(imageStruct.roofTypes(1).colorRegion(4).
   filteredMask.RGB.filteredMeanStdMask,imageStruct.roofTypes(1).
   colorRegion(4).filteredMask.HSV.filteredMeanStdMask);
imageStruct.roofTypes(1).colorRegion(4).filteredMask.total.meanStd.
   OR = imageStruct.roofTypes(1).colorRegion(4).filteredMask.RGB.
   filteredMeanStdMask|imageStruct.roofTypes(1).colorRegion(4).
   filteredMask.HSV.filteredMeanStdMask;
```

```
%% Trials I. Filter individual masks. Red (Shadow)
imageStruct.roofTypes(1).colorRegion(3).filteredMask = struct();

imageStruct.roofTypes(1).colorRegion(3).filteredMask.RGB = struct()
   ;
imageStruct.roofTypes(1).colorRegion(3).filteredMask.RGB.
   filteredMeanStdMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(3).meanStdMask.RGB,200,800);
imageStruct.roofTypes(1).colorRegion(3).filteredMask.RGB.
   filteredMinMaxMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(3).meanStdMask.RGB,200,800);

imageStruct.roofTypes(1).colorRegion(3).filteredMask.HSV = struct()
   ;
imageStruct.roofTypes(1).colorRegion(3).filteredMask.HSV.
   filteredMeanStdMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(3).meanStdMask.HSV,200,800);
imageStruct.roofTypes(1).colorRegion(3).filteredMask.HSV.
   filteredMinMaxMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(3).minMaxMask.HSV,200,800);

% Blob OR
imageStruct.roofTypes(1).colorRegion(3).filteredMask.total = struct
   ();
imageStruct.roofTypes(1).colorRegion(3).filteredMask.total.RGB =
   struct();
imageStruct.roofTypes(1).colorRegion(3).filteredMask.total.RGB.
   blobOR = blobOR(imageStruct.roofTypes(1).colorRegion(3).
   filteredMask.RGB.filteredMinMaxMask,imageStruct.roofTypes(1).
   colorRegion(3).filteredMask.RGB.filteredMeanStdMask);
imageStruct.roofTypes(1).colorRegion(3).filteredMask.total.RGB.OR =
    imageStruct.roofTypes(1).colorRegion(3).filteredMask.RGB.
   filteredMinMaxMask|imageStruct.roofTypes(1).colorRegion(3).
   filteredMask.RGB.filteredMeanStdMask;
imageStruct.roofTypes(1).colorRegion(3).filteredMask.total.HSV =
   struct();
imageStruct.roofTypes(1).colorRegion(3).filteredMask.total.HSV.
   blobOR = blobOR(imageStruct.roofTypes(1).colorRegion(3).
   filteredMask.HSV.filteredMinMaxMask,imageStruct.roofTypes(1).
   colorRegion(3).filteredMask.HSV.filteredMeanStdMask);
imageStruct.roofTypes(1).colorRegion(3).filteredMask.total.HSV.OR =
    imageStruct.roofTypes(1).colorRegion(3).filteredMask.HSV.
   filteredMinMaxMask|imageStruct.roofTypes(1).colorRegion(3).
   filteredMask.HSV.filteredMeanStdMask;
```

```
%% Trials I. Filter individual masks. Orange (Shadow)

imageStruct.roofTypes(1).colorRegion(2).filteredMask = struct();

imageStruct.roofTypes(1).colorRegion(2).filteredMask.RGB = struct()
   ;
imageStruct.roofTypes(1).colorRegion(2).filteredMask.RGB.
   filteredMeanStdMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(2).meanStdMask.RGB,200,800);
imageStruct.roofTypes(1).colorRegion(2).filteredMask.RGB.
   filteredMinMaxMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(2).meanStdMask.RGB,200,800);

imageStruct.roofTypes(1).colorRegion(2).filteredMask.HSV = struct()
   ;
imageStruct.roofTypes(1).colorRegion(2).filteredMask.HSV.
   filteredMeanStdMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(2).meanStdMask.HSV,200,800);
imageStruct.roofTypes(1).colorRegion(2).filteredMask.HSV.
   filteredMinMaxMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(2).minMaxMask.HSV,200,800);

% Blob OR
imageStruct.roofTypes(1).colorRegion(2).filteredMask.total = struct
   ();
imageStruct.roofTypes(1).colorRegion(2).filteredMask.total.RGB =
   struct();
imageStruct.roofTypes(1).colorRegion(2).filteredMask.total.RGB.
   blobOR = blobOR(imageStruct.roofTypes(1).colorRegion(2).
   filteredMask.RGB.filteredMinMaxMask,imageStruct.roofTypes(1).
   colorRegion(2).filteredMask.RGB.filteredMeanStdMask);
imageStruct.roofTypes(1).colorRegion(2).filteredMask.total.RGB.OR =
    imageStruct.roofTypes(1).colorRegion(2).filteredMask.RGB.
   filteredMinMaxMask|imageStruct.roofTypes(1).colorRegion(2).
   filteredMask.RGB.filteredMeanStdMask;
imageStruct.roofTypes(1).colorRegion(2).filteredMask.total.HSV =
   struct();
imageStruct.roofTypes(1).colorRegion(2).filteredMask.total.HSV.
   blobOR = blobOR(imageStruct.roofTypes(1).colorRegion(2).
   filteredMask.HSV.filteredMinMaxMask,imageStruct.roofTypes(1).
   colorRegion(2).filteredMask.HSV.filteredMeanStdMask);
imageStruct.roofTypes(1).colorRegion(2).filteredMask.total.HSV.OR =
    imageStruct.roofTypes(1).colorRegion(2).filteredMask.HSV.
   filteredMinMaxMask|imageStruct.roofTypes(1).colorRegion(2).
   filteredMask.HSV.filteredMeanStdMask;
```

```
%% Trials I. Filter individual masks. darkRed (Light)

imageStruct.roofTypes(1).colorRegion(5).filteredMask = struct();

imageStruct.roofTypes(1).colorRegion(5).filteredMask.RGB = struct()
   ;
imageStruct.roofTypes(1).colorRegion(5).filteredMask.RGB.
   filteredMeanStdMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(5).meanStdMask.RGB,400,1200);
imageStruct.roofTypes(1).colorRegion(5).filteredMask.RGB.
   filteredMinMaxMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(5).meanStdMask.RGB,400,1200);

imageStruct.roofTypes(1).colorRegion(5).filteredMask.HSV = struct()
   ;
imageStruct.roofTypes(1).colorRegion(5).filteredMask.HSV.
   filteredMeanStdMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(5).meanStdMask.HSV,400,1200);
imageStruct.roofTypes(1).colorRegion(5).filteredMask.HSV.
   filteredMinMaxMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(5).minMaxMask.HSV,400,1200);

% Blob OR
imageStruct.roofTypes(1).colorRegion(5).filteredMask.total = struct
   ();
imageStruct.roofTypes(1).colorRegion(5).filteredMask.total.minMax =
    struct();
imageStruct.roofTypes(1).colorRegion(5).filteredMask.total.minMax.
   blobOR = blobOR(imageStruct.roofTypes(1).colorRegion(5).
   filteredMask.RGB.filteredMinMaxMask,imageStruct.roofTypes(1).
   colorRegion(5).filteredMask.HSV.filteredMinMaxMask);
imageStruct.roofTypes(1).colorRegion(5).filteredMask.total.minMax.
   OR = imageStruct.roofTypes(1).colorRegion(5).filteredMask.RGB.
   filteredMinMaxMask|imageStruct.roofTypes(1).colorRegion(5).
   filteredMask.HSV.filteredMinMaxMask;
imageStruct.roofTypes(1).colorRegion(5).filteredMask.total.meanStd
   = struct();
imageStruct.roofTypes(1).colorRegion(5).filteredMask.total.meanStd.
   blobOR = blobOR(imageStruct.roofTypes(1).colorRegion(5).
   filteredMask.RGB.filteredMeanStdMask,imageStruct.roofTypes(1).
   colorRegion(5).filteredMask.HSV.filteredMeanStdMask);
```

```
imageStruct.roofTypes(1).colorRegion(5).filteredMask.total.meanStd.
   OR = imageStruct.roofTypes(1).colorRegion(5).filteredMask.RGB.
   filteredMeanStdMask|imageStruct.roofTypes(1).colorRegion(5).
   filteredMask.HSV.filteredMeanStdMask;

%% Trials I. Filter individual masks. DarkBlue (Light)

imageStruct.roofTypes(1).colorRegion(6).filteredMask = struct();

imageStruct.roofTypes(1).colorRegion(6).filteredMask.RGB = struct()
   ;
imageStruct.roofTypes(1).colorRegion(6).filteredMask.RGB.
   filteredMeanStdMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(6).meanStdMask.RGB,400,1200);
imageStruct.roofTypes(1).colorRegion(6).filteredMask.RGB.
   filteredMinMaxMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(6).meanStdMask.RGB,400,1200);

imageStruct.roofTypes(1).colorRegion(6).filteredMask.HSV = struct()
   ;
imageStruct.roofTypes(1).colorRegion(6).filteredMask.HSV.
   filteredMeanStdMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(6).meanStdMask.HSV,400,1200);
imageStruct.roofTypes(1).colorRegion(6).filteredMask.HSV.
   filteredMinMaxMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(6).minMaxMask.HSV,400,1200);

% Blob OR
imageStruct.roofTypes(1).colorRegion(6).filteredMask.total = struct
   ();
imageStruct.roofTypes(1).colorRegion(6).filteredMask.total.minMax =
    struct();
imageStruct.roofTypes(1).colorRegion(6).filteredMask.total.minMax.
   blobOR = blobOR(imageStruct.roofTypes(1).colorRegion(6).
   filteredMask.RGB.filteredMinMaxMask,imageStruct.roofTypes(1).
   colorRegion(6).filteredMask.HSV.filteredMinMaxMask);
imageStruct.roofTypes(1).colorRegion(6).filteredMask.total.minMax.
   OR = imageStruct.roofTypes(1).colorRegion(6).filteredMask.RGB.
   filteredMinMaxMask|imageStruct.roofTypes(1).colorRegion(6).
   filteredMask.HSV.filteredMinMaxMask;
imageStruct.roofTypes(1).colorRegion(6).filteredMask.total.meanStd
   = struct();
```

```
imageStruct.roofTypes(1).colorRegion(6).filteredMask.total.meanStd.
   blobOR = blobOR(imageStruct.roofTypes(1).colorRegion(6).
   filteredMask.RGB.filteredMeanStdMask,imageStruct.roofTypes(1).
   colorRegion(6).filteredMask.HSV.filteredMeanStdMask);
imageStruct.roofTypes(1).colorRegion(6).filteredMask.total.meanStd.
   OR = imageStruct.roofTypes(1).colorRegion(6).filteredMask.RGB.
   filteredMeanStdMask|imageStruct.roofTypes(1).colorRegion(6).
   filteredMask.HSV.filteredMeanStdMask;

%% Trials I. Filter individual masks. DarkBlue (Shadow)

imageStruct.roofTypes(1).colorRegion(7).filteredMask = struct();

imageStruct.roofTypes(1).colorRegion(7).filteredMask.RGB = struct()
   ;
imageStruct.roofTypes(1).colorRegion(7).filteredMask.RGB.
   filteredMeanStdMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(7).meanStdMask.RGB,200,800);
imageStruct.roofTypes(1).colorRegion(7).filteredMask.RGB.
   filteredMinMaxMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(7).meanStdMask.RGB,200,800);

imageStruct.roofTypes(1).colorRegion(7).filteredMask.HSV = struct()
   ;
imageStruct.roofTypes(1).colorRegion(7).filteredMask.HSV.
   filteredMeanStdMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(7).meanStdMask.HSV,200,800);
imageStruct.roofTypes(1).colorRegion(7).filteredMask.HSV.
   filteredMinMaxMask = filterIndividualMask(imageStruct.roofTypes
   (1).colorRegion(7).minMaxMask.HSV,200,800);

% Blob OR
imageStruct.roofTypes(1).colorRegion(7).filteredMask.total = struct
   ();
imageStruct.roofTypes(1).colorRegion(7).filteredMask.total.RGB =
   struct();
imageStruct.roofTypes(1).colorRegion(7).filteredMask.total.RGB.
   blobOR = blobOR(imageStruct.roofTypes(1).colorRegion(7).
   filteredMask.RGB.filteredMinMaxMask,imageStruct.roofTypes(1).
   colorRegion(7).filteredMask.RGB.filteredMeanStdMask);
imageStruct.roofTypes(1).colorRegion(7).filteredMask.total.RGB.OR =
    imageStruct.roofTypes(1).colorRegion(7).filteredMask.RGB.
   filteredMinMaxMask|imageStruct.roofTypes(1).colorRegion(7).
   filteredMask.RGB.filteredMeanStdMask;
```

```
imageStruct.roofTypes(1).colorRegion(7).filteredMask.total.HSV =
   struct();
imageStruct.roofTypes(1).colorRegion(7).filteredMask.total.HSV.
   blobOR = blobOR(imageStruct.roofTypes(1).colorRegion(7).
   filteredMask.HSV.filteredMinMaxMask,imageStruct.roofTypes(1).
   colorRegion(7).filteredMask.HSV.filteredMeanStdMask);
imageStruct.roofTypes(1).colorRegion(7).filteredMask.total.HSV.OR =
    imageStruct.roofTypes(1).colorRegion(7).filteredMask.HSV.
   filteredMinMaxMask|imageStruct.roofTypes(1).colorRegion(7).
   filteredMask.HSV.filteredMeanStdMask;


%% Trials I. Fuse all the masks (LIGHT)

imageStruct.roofTypes(1).mask.fusedFilteredMask = struct();
imageStruct.roofTypes(1).mask.fusedFilteredMask.light = struct();

imageStruct.roofTypes(1).mask.fusedFilteredMask.light.minMax =
   struct();
imageStruct.roofTypes(1).mask.fusedFilteredMask.light.minMax.OR =
   imageStruct.roofTypes(1).colorRegion(4).filteredMask.total.
   minMax.OR|imageStruct.roofTypes(1).colorRegion(1).filteredMask.
   total.minMax.OR;
imageStruct.roofTypes(1).mask.fusedFilteredMask.light.minMax.blobOR
    = imageStruct.roofTypes(1).colorRegion(4).filteredMask.total.
   minMax.blobOR|imageStruct.roofTypes(1).colorRegion(1).
   filteredMask.total.minMax.blobOR;

imageStruct.roofTypes(1).mask.fusedFilteredMask.light.meanStd =
   struct();
imageStruct.roofTypes(1).mask.fusedFilteredMask.light.meanStd.OR =
   imageStruct.roofTypes(1).colorRegion(4).filteredMask.total.
   meanStd.OR|imageStruct.roofTypes(1).colorRegion(1).filteredMask.
   total.meanStd.OR;
imageStruct.roofTypes(1).mask.fusedFilteredMask.light.meanStd.
   blobOR = imageStruct.roofTypes(1).colorRegion(4).filteredMask.
   total.meanStd.blobOR|imageStruct.roofTypes(1).colorRegion(1).
   filteredMask.total.meanStd.blobOR;

%% Trials I. OPTION 2. Fuse all the masks (LIGHT) + DarkRed

imageStruct.roofTypes(1).mask.fusedFilteredMask = struct();
imageStruct.roofTypes(1).mask.fusedFilteredMask.light = struct();
```

```
imageStruct.roofTypes(1).mask.fusedFilteredMask.light.minMax =
   struct();
imageStruct.roofTypes(1).mask.fusedFilteredMask.light.minMax.OR =
   imageStruct.roofTypes(1).colorRegion(4).filteredMask.total.
   minMax.OR|imageStruct.roofTypes(1).colorRegion(1).filteredMask.
   total.minMax.OR|imageStruct.roofTypes(1).colorRegion(5).
   filteredMask.total.minMax.OR;
imageStruct.roofTypes(1).mask.fusedFilteredMask.light.minMax.blobOR
    = imageStruct.roofTypes(1).colorRegion(4).filteredMask.total.
   minMax.blobOR|imageStruct.roofTypes(1).colorRegion(1).
   filteredMask.total.minMax.blobOR|imageStruct.roofTypes(1).
   colorRegion(5).filteredMask.total.minMax.blobOR;

imageStruct.roofTypes(1).mask.fusedFilteredMask.light.meanStd =
   struct();
imageStruct.roofTypes(1).mask.fusedFilteredMask.light.meanStd.OR =
   imageStruct.roofTypes(1).colorRegion(4).filteredMask.total.
   meanStd.OR|imageStruct.roofTypes(1).colorRegion(1).filteredMask.
   total.meanStd.OR|imageStruct.roofTypes(1).colorRegion(5).
   filteredMask.total.meanStd.OR;
imageStruct.roofTypes(1).mask.fusedFilteredMask.light.meanStd.
   blobOR = imageStruct.roofTypes(1).colorRegion(4).filteredMask.
   total.meanStd.blobOR|imageStruct.roofTypes(1).colorRegion(1).
   filteredMask.total.meanStd.blobOR|imageStruct.roofTypes(1).
   colorRegion(5).filteredMask.total.meanStd.blobOR;


%% Trials I. OPTION 3. Fuse all the masks (LIGHT) + DarkBlue

imageStruct.roofTypes(1).mask.fusedFilteredMask = struct();
imageStruct.roofTypes(1).mask.fusedFilteredMask.light = struct();

imageStruct.roofTypes(1).mask.fusedFilteredMask.light.minMax =
   struct();
imageStruct.roofTypes(1).mask.fusedFilteredMask.light.minMax.OR =
   imageStruct.roofTypes(1).colorRegion(4).filteredMask.total.
   minMax.OR|imageStruct.roofTypes(1).colorRegion(1).filteredMask.
   total.minMax.OR|imageStruct.roofTypes(1).colorRegion(6).
   filteredMask.total.minMax.OR;
imageStruct.roofTypes(1).mask.fusedFilteredMask.light.minMax.blobOR
    = imageStruct.roofTypes(1).colorRegion(4).filteredMask.total.
   minMax.blobOR|imageStruct.roofTypes(1).colorRegion(1).
   filteredMask.total.minMax.blobOR|imageStruct.roofTypes(1).
   colorRegion(6).filteredMask.total.minMax.blobOR;
```

```
imageStruct.roofTypes(1).mask.fusedFilteredMask.light.meanStd =
   struct();
imageStruct.roofTypes(1).mask.fusedFilteredMask.light.meanStd.OR =
   imageStruct.roofTypes(1).colorRegion(4).filteredMask.total.
   meanStd.OR|imageStruct.roofTypes(1).colorRegion(1).filteredMask.
   total.meanStd.OR|imageStruct.roofTypes(1).colorRegion(6).
   filteredMask.total.meanStd.OR;
imageStruct.roofTypes(1).mask.fusedFilteredMask.light.meanStd.
   blobOR = imageStruct.roofTypes(1).colorRegion(4).filteredMask.
   total.meanStd.blobOR|imageStruct.roofTypes(1).colorRegion(1).
   filteredMask.total.meanStd.blobOR|imageStruct.roofTypes(1).
   colorRegion(6).filteredMask.total.meanStd.blobOR;

%% Trials I. Fuse all the masks (SHADOW)

imageStruct.roofTypes(1).mask.fusedFilteredMask.shadow = struct();

imageStruct.roofTypes(1).mask.fusedFilteredMask.shadow.RGB = struct
   ();
imageStruct.roofTypes(1).mask.fusedFilteredMask.shadow.RGB.OR =
   imageStruct.roofTypes(1).colorRegion(2).filteredMask.total.RGB.
   OR|imageStruct.roofTypes(1).colorRegion(3).filteredMask.total.
   RGB.OR;
imageStruct.roofTypes(1).mask.fusedFilteredMask.shadow.RGB.blobOR =
    imageStruct.roofTypes(1).colorRegion(2).filteredMask.total.RGB.
   blobOR|imageStruct.roofTypes(1).colorRegion(3).filteredMask.
   total.RGB.blobOR;

imageStruct.roofTypes(1).mask.fusedFilteredMask.shadow.HSV = struct
   ();
imageStruct.roofTypes(1).mask.fusedFilteredMask.shadow.HSV.OR =
   imageStruct.roofTypes(1).colorRegion(2).filteredMask.total.HSV.
   OR|imageStruct.roofTypes(1).colorRegion(3).filteredMask.total.
   HSV.OR;
imageStruct.roofTypes(1).mask.fusedFilteredMask.shadow.HSV.blobOR =
    imageStruct.roofTypes(1).colorRegion(2).filteredMask.total.HSV.
   blobOR|imageStruct.roofTypes(1).colorRegion(3).filteredMask.
   total.HSV.blobOR;

%% Trials I. OPTION 3. Fuse all the masks (SHADOW) + DarkBlue

imageStruct.roofTypes(1).mask.fusedFilteredMask.shadow = struct();

imageStruct.roofTypes(1).mask.fusedFilteredMask.shadow.RGB = struct
   ();
```

```
imageStruct.roofTypes(1).mask.fusedFilteredMask.shadow.RGB.OR =
   imageStruct.roofTypes(1).colorRegion(2).filteredMask.total.RGB.
   OR|imageStruct.roofTypes(1).colorRegion(3).filteredMask.total.
   RGB.OR|imageStruct.roofTypes(1).colorRegion(7).filteredMask.
   total.RGB.OR;
imageStruct.roofTypes(1).mask.fusedFilteredMask.shadow.RGB.blobOR =
    imageStruct.roofTypes(1).colorRegion(2).filteredMask.total.RGB.
   blobOR|imageStruct.roofTypes(1).colorRegion(3).filteredMask.
   total.RGB.blobOR|imageStruct.roofTypes(1).colorRegion(7).
   filteredMask.total.RGB.blobOR;

imageStruct.roofTypes(1).mask.fusedFilteredMask.shadow.HSV = struct
   ();
imageStruct.roofTypes(1).mask.fusedFilteredMask.shadow.HSV.OR =
   imageStruct.roofTypes(1).colorRegion(2).filteredMask.total.HSV.
   OR|imageStruct.roofTypes(1).colorRegion(3).filteredMask.total.
   HSV.OR|imageStruct.roofTypes(1).colorRegion(7).filteredMask.
   total.HSV.OR;
imageStruct.roofTypes(1).mask.fusedFilteredMask.shadow.HSV.blobOR =
    imageStruct.roofTypes(1).colorRegion(2).filteredMask.total.HSV.
   blobOR|imageStruct.roofTypes(1).colorRegion(3).filteredMask.
   total.HSV.blobOR|imageStruct.roofTypes(1).colorRegion(7).
   filteredMask.total.HSV.blobOR;


%% Trials I. Fuse light and shadow

% Fuse Light
imageStruct.roofTypes(1).mask.fusedFilteredMask.light.total.blobOR
   = blobOR(blobOR(imageStruct.roofTypes(1).mask.fusedFilteredMask.
   light.meanStd.blobOR,imageStruct.roofTypes(1).mask.
   fusedFilteredMask.light.meanStd.OR),...
blobOR(imageStruct.roofTypes(1).mask.fusedFilteredMask.light.minMax
   .blobOR,imageStruct.roofTypes(1).mask.fusedFilteredMask.light.
   minMax.OR));

% Fuse Shadow n Light
imageStruct.roofTypes(1).mask.fusedFilteredMask.total = struct();
imageStruct.roofTypes(1).mask.fusedFilteredMask.total = imageStruct
   .roofTypes(1).mask.fusedFilteredMask.shadow.HSV.OR|imageStruct.
   roofTypes(1).mask.fusedFilteredMask.light.total.blobOR;

%% Trials I. Crop blobs
croppedMask = cropBlobs(imageStruct.roofTypes(1).mask.
   fusedFilteredMask.total);
```

```
%% More cropping if necessary
croppedMask = cropBlobs(croppedMask);

%% Visualize it
visualizeMask(imageStruct.image.RGB,croppedMask);

%% Finally close the mask

% First, fuse super close blobs.
imageStruct.roofTypes(1).mask.finalMask = myFill(myClose(
   croppedMask,ones(3),3,3),2000);
visualizeMask(imageStruct.image.RGB,imageStruct.roofTypes(1).mask.
   finalMask);

% Second, fuse individual blobs.
lastLabels = bwlabel(imageStruct.roofTypes(1).mask.finalMask);
blobs = unique(lastLabels(:));

finalBlobs = zeros(size(lastLabels));
for i = blobs(blobs>0)'
    blobMask = ismember(lastLabels,i);
    blobClosedMask = myClose(blobMask,ones(3),4,4);
    finalBlobs = finalBlobs|blobClosedMask;
end

imageStruct.roofTypes(1).mask.finalMask = myFill(finalBlobs,2000);
   % This brings problems


%% Visualiza it

visualizeMask(imageStruct.image.RGB,imageStruct.roofTypes(1).mask.
   finalMask);

%% Trials I. Calculate the area of each blob

blobLabels = bwlabel(imageStruct.roofTypes(1).mask.finalMask);
blobProps = regionprops(blobLabels,"Area");
blobAreas = [blobProps.Area]/(imageStruct.image.tiff.getTag("
   XResolution")*imageStruct.image.tiff.getTag("YResolution"));
imageStruct.roofTypes(1).area = struct();
imageStruct.roofTypes(1).area.totalBlobArea = sum(blobAreas);

% Threshold areas lower than 50,75,100m2
```

```
imageStruct.roofTypes(1).mask.thresholdedMask = struct();

k = find(blobAreas>50);
imageStruct.roofTypes(1).mask.thresholdedMask.fifty = ismember(
   blobLabels,k);
imageStruct.roofTypes(1).area.fifty = sum(blobAreas(k));
figure("Name","Threshold50");
imshow(imageStruct.roofTypes(1).mask.thresholdedMask.fifty);
k = find(blobAreas>75);
imageStruct.roofTypes(1).mask.thresholdedMask.seventyFive =
   ismember(blobLabels,k);
imageStruct.roofTypes(1).area.seventyFive = sum(blobAreas(k));
figure("Name","Threshold75")
imshow(imageStruct.roofTypes(1).mask.thresholdedMask.seventyFive);
k = find(blobAreas>100);
imageStruct.roofTypes(1).mask.thresholdedMask.hundred = ismember(
   blobLabels,k);
imageStruct.roofTypes(1).area.hundred = sum(blobAreas(k));
figure("Name","Threshold100")
imshow(imageStruct.roofTypes(1).mask.thresholdedMask.hundred);

%% Sum area to rooftype sum. Initialise

roofType(1).area = struct();
roofType(1).area.LIllaPerduda = struct();
roofType(1).area.LIllaPerduda.totalBlobArea = 0;
roofType(1).area.LIllaPerduda.fifty = 0;
roofType(1).area.LIllaPerduda.seventyFive = 0;
roofType(1).area.LIllaPerduda.hundred = 0;

%% Sum area to rooftype sum. Sum

roofType(1).area.LIllaPerduda.totalBlobArea = roofType(1).area.
   LIllaPerduda.totalBlobArea + imageStruct.roofTypes(1).area.
   totalBlobArea;
roofType(1).area.LIllaPerduda.fifty = roofType(1).area.LIllaPerduda
   .fifty + imageStruct.roofTypes(1).area.fifty;
roofType(1).area.LIllaPerduda.seventyFive = roofType(1).area.
   LIllaPerduda.seventyFive + imageStruct.roofTypes(1).area.
   seventyFive;
roofType(1).area.LIllaPerduda.hundred = roofType(1).area.
   LIllaPerduda.hundred + imageStruct.roofTypes(1).area.hundred;

%% Trials I. Save the struct
```

```
save("MATLABData\imageMasks\"+imageStruct.name,"imageStruct");

%% Trials I. Mask histogram

mask = imageStruct.roofTypes(1).mask.finalMask;

%Pixel Values
idx = find(mask);
imageH = imageStruct.image.HSV(:,:,1);
imageS = imageStruct.image.HSV(:,:,2);
imageV = imageStruct.image.HSV(:,:,3);
maskPixelHSV = [imageH(idx), imageS(idx), imageV(idx)];

imageR = imageStruct.image.RGB(:,:,1);
imageG = imageStruct.image.RGB(:,:,2);
imageB = imageStruct.image.RGB(:,:,3);
maskPixelRGB = [imageR(idx), imageG(idx), imageB(idx)];


figure("Name","Roof histogram RGB");
subplot(1,3,1);
binscatter(maskPixelRGB(:,1),maskPixelRGB(:,2))
title("Red vs green scatter","Interpreter","latex");
xlabel("Red","Interpreter","Latex");
ylabel("Green","Interpreter","latex");
xlim([0 255]);
ylim([0 255]);
subplot(1,3,2);
binscatter(maskPixelRGB(:,1),maskPixelRGB(:,3))
title("Red vs blue scatter","Interpreter","latex");
xlabel("Red","Interpreter","Latex");
ylabel("Blue","Interpreter","latex");
xlim([0 255]);
ylim([0 255]);
subplot(1,3,3);
binscatter(maskPixelRGB(:,2),maskPixelRGB(:,3))
title("Green vs blue scatter","Interpreter","latex");
xlabel("Green","Interpreter","Latex");
ylabel("Blue","Interpreter","latex");
xlim([0 1]);
ylim([0 1]);

figure("Name","Roof histogram HSV");
subplot(1,3,1);
binscatter(maskPixelHSV(:,1),maskPixelHSV(:,2))
```

```
title("Hue vs saturation scatter","Interpreter","latex");
xlabel("Hue","Interpreter","Latex");
ylabel("Saturation","Interpreter","latex");
xlim([0 1]);
ylim([0 1]);
subplot(1,3,2);
binscatter(maskPixelHSV(:,1),maskPixelHSV(:,3))
title("Hue vs value scatter","Interpreter","latex");
xlabel("Hue","Interpreter","Latex");
ylabel("Value","Interpreter","latex");
xlim([0 1]);
ylim([0 1]);
subplot(1,3,3);
binscatter(maskPixelHSV(:,2),maskPixelHSV(:,3))
title("Saturation vs value scatter","Interpreter","latex");
xlabel("Saturation","Interpreter","Latex");
ylabel("Value","Interpreter","latex");
xlim([0 1]);
ylim([0 1]);

%% Trials I. kmeans on pixelValues

kmeanHSVIdx = kmeans(maskPixelHSV,3,"MaxIter",200);
newMaskHSV = zeros(size(mask));
newMaskHSV(idx) = kmeanHSVIdx;
figure("Name","kmeans separation on HSV mask");
imshow(newMaskHSV,[]);

kmeanRGBIdx = kmeans(double(maskPixelRGB),3,"MaxIter",200);
newMaskRGB = zeros(size(mask));
newMaskRGB(idx) = kmeanRGBIdx;
figure("Name","kmeans separation on RGB mask");
imshow(newMaskRGB,[]);
%% Trials II. Gradient

[gR,gDirR] = imgradient(imageStruct.image.RGB(:,:,1),'
   CentralDifference');
[gG,gDirG] = imgradient(imageStruct.image.RGB(:,:,2),'
   CentralDifference');
[gB,gDirB] = imgradient(imageStruct.image.RGB(:,:,3),'
   CentralDifference');

gImage = cat(3,rescale(gR),rescale(gG),rescale(gB));
gImageHSV = rgb2hsv(gImage);
gMagnitude = rescale(gR.^2+gG.^2+gB.^2);
```

```
figure('Name',"3-Color Gradient");
imshow(gImage);
figure('Name',"3-Color Gradient Value");
imshow(gImageHSV(:,:,3));
figure('Name',"3-Color Gradient Magnitude");
imshow(gMagnitude);

%% Trials II. Gradient mask

gradientMask = imbinarize(gImageHSV(:,:,3));
figure("Name","Border Mask")
imshow(gradientMask);

%% Trials II. Filtering gradient mask

closedG = myClose(gradientMask,[1 1 1; 1 1 1; 1 1 1],2,2);
figure("Name","Closed border Mask")
imshow(closedG);

homogenRegions = 1-closedG;
figure("Name","Closed homogen regions mask")
imshow(homogenRegions);

%% Trials III. Kmeans
imageH = imageStruct.image.medianFiltered.HSV(:,:,1);
imageS = imageStruct.image.medianFiltered.HSV(:,:,2);
imageV = imageStruct.image.medianFiltered.HSV(:,:,3);

imagePixels = [imageH(:), imageS(:), imageV(:)];

%evaluation = evalclusters(imagePixels,"kmeans","silhouette","KList
   ",1:4);

idx = kmeans(imagePixels,10,"MaxIter",100);
kmeansMask = reshape(idx,size(AerialImage(:,:,1)));
figure("Name","kmean separartion");
imshow(kmeansMask,[]);
%% Functions

function pClicks = selectRoofPixels(Image)

    f = figure('Name','Selected building pixeels');
    imshow(Image);
    hold on
```

```
    pClicks = zeros(size(Image,[1 2]));
    pC=imshow(pClicks);
    pC.AlphaData = 0.4;

    key = 0;

    while key ~= 13

        % Wait until some key or button is pressed and get the
           position of the
        % mouse and the pressed key.
        [x,y,key] = ginput(1);
        xPixel = round(x);
        yPixel = round(y);

        % If it was not enter
        if key ~= 13
            % If it was the left click add that FA to the deletion
               list.
            if key == 1
                pClicks(yPixel,xPixel) = 1;
                % If it was the right click remove the FA from the
                   deletion
                % list.
            elseif key == 3
                pClicks(yPixel,xPixel) = 0;
            end
        end

        % Refresh the image and show the selected FAs in red.
        figure(f);
        delete(pC);
        pC=imshow(conv2(pClicks,ones(5),"same")>0);
        pC.AlphaData = 0.4;

    end

end

function pixelValues = getPixelValues(Image,selectedPixels)

    [yPixel,xPixel] = find(selectedPixels);
    yPixelcoord = repelem(yPixel,3);
    xPixelcoord = repelem(xPixel,3);
    zPixelcoord = repmat((1:3)',length(xPixel),1);
```

```
    PixelIdx = sub2ind(size(Image),yPixelcoord,xPixelcoord,
       zPixelcoord);
    pixelValues = reshape(Image(PixelIdx),[3,length(xPixel),])';

end

function regionStats = getStatistics(colorRegion)

    regionStats.RGB = struct();
    regionStats.RGB.min = min(colorRegion.pixelValuesRGB);
    regionStats.RGB.max = max(colorRegion.pixelValuesRGB);
    regionStats.RGB.mean = mean(colorRegion.pixelValuesRGB);
    regionStats.RGB.std = std(double(colorRegion.pixelValuesRGB));

    regionStats.HSV = struct();
    regionStats.HSV.min = min(colorRegion.pixelValuesHSV);
    regionStats.HSV.max = max(colorRegion.pixelValuesHSV);
    regionStats.HSV.mean = mean(colorRegion.pixelValuesHSV);
    regionStats.HSV.std = std(colorRegion.pixelValuesHSV);

end

function IS = createMask(IS,RT,idx,medianFilt)

    if medianFilt
        imageHSV = IS.image.medianFiltered.HSV;
        imageRGB = IS.image.medianFiltered.RGB;
    else
        imageHSV = IS.image.HSV;
        imageHSV = IS.image.RGB;
    end

    nColorRegion = length(RT(idx).colorRegions);

    imageH = IS.image.HSV(:,:,1);
    imageS = IS.image.HSV(:,:,2);
    imageV = IS.image.HSV(:,:,3);

    imageR = IS.image.RGB(:,:,1);
    imageG = IS.image.RGB(:,:,2);
    imageB = IS.image.RGB(:,:,3);

    for i = 1:nColorRegion

        colorRegion = RT(idx).colorRegions(i);
```

```
%minMax mask

minMaxRoofMaskH = imageH < colorRegion.statistics.HSV.max
   (1) & imageH > colorRegion.statistics.HSV.min(1);
minMaxRoofMaskS = imageS < colorRegion.statistics.HSV.max
   (2) & imageS > colorRegion.statistics.HSV.min(2);
minMaxRoofMaskV = imageV < colorRegion.statistics.HSV.max
   (3) & imageV > colorRegion.statistics.HSV.min(3);

minMaxRoofMaskR = imageR < colorRegion.statistics.RGB.max
   (1) & imageR > colorRegion.statistics.RGB.min(1);
minMaxRoofMaskG = imageG < colorRegion.statistics.RGB.max
   (2) & imageG > colorRegion.statistics.RGB.min(2);
minMaxRoofMaskB = imageB < colorRegion.statistics.RGB.max
   (3) & imageB > colorRegion.statistics.RGB.min(3);

IS.roofTypes(idx).colorRegion(i).minMaxMask = struct();
IS.roofTypes(idx).colorRegion(i).minMaxMask.R =
   minMaxRoofMaskR;
IS.roofTypes(idx).colorRegion(i).minMaxMask.G =
   minMaxRoofMaskG;
IS.roofTypes(idx).colorRegion(i).minMaxMask.B =
   minMaxRoofMaskB;
IS.roofTypes(idx).colorRegion(i).minMaxMask.RGB =
   minMaxRoofMaskR & minMaxRoofMaskG & minMaxRoofMaskB;

IS.roofTypes(idx).colorRegion(i).minMaxMask.H =
   minMaxRoofMaskH;
IS.roofTypes(idx).colorRegion(i).minMaxMask.S =
   minMaxRoofMaskS;
IS.roofTypes(idx).colorRegion(i).minMaxMask.V =
   minMaxRoofMaskV;
IS.roofTypes(idx).colorRegion(i).minMaxMask.HSV =
   minMaxRoofMaskH & minMaxRoofMaskS & minMaxRoofMaskV;

IS.roofTypes(idx).colorRegion(i).minMaxMask.total = IS.
   roofTypes(idx).colorRegion(i).minMaxMask.RGB & ...
    IS.roofTypes(idx).colorRegion(i).minMaxMask.HSV;

% meanStd mask
```

```
meanStdRoofMaskH = imageH < colorRegion.statistics.HSV.mean
   (1)+3*colorRegion.statistics.HSV.std(1) & imageH >
   colorRegion.statistics.HSV.mean(1)-3*colorRegion.
   statistics.HSV.std(1);
meanStdRoofMaskS = imageS < colorRegion.statistics.HSV.mean
   (2)+3*colorRegion.statistics.HSV.std(2) & imageS >
   colorRegion.statistics.HSV.mean(2)-3*colorRegion.
   statistics.HSV.std(2);
meanStdRoofMaskV = imageV < colorRegion.statistics.HSV.mean
   (3)+3*colorRegion.statistics.HSV.std(3) & imageV >
   colorRegion.statistics.HSV.mean(3)-3*colorRegion.
   statistics.HSV.std(3);

meanStdRoofMaskR = imageR < colorRegion.statistics.RGB.mean
   (1)+3*colorRegion.statistics.RGB.std(1) & imageR >
   colorRegion.statistics.RGB.mean(1)-3*colorRegion.
   statistics.RGB.std(1);
meanStdRoofMaskG = imageG < colorRegion.statistics.RGB.mean
   (2)+3*colorRegion.statistics.RGB.std(2) & imageG >
   colorRegion.statistics.RGB.mean(2)-3*colorRegion.
   statistics.RGB.std(2);
meanStdRoofMaskB = imageB < colorRegion.statistics.RGB.mean
   (3)+3*colorRegion.statistics.RGB.std(3) & imageB >
   colorRegion.statistics.RGB.mean(3)-3*colorRegion.
   statistics.RGB.std(3);

IS.roofTypes(idx).colorRegion(i).meanStdMask = struct();
IS.roofTypes(idx).colorRegion(i).meanStdMask.R =
   meanStdRoofMaskR;
IS.roofTypes(idx).colorRegion(i).meanStdMask.G =
   meanStdRoofMaskG;
IS.roofTypes(idx).colorRegion(i).meanStdMask.B =
   meanStdRoofMaskB;
IS.roofTypes(idx).colorRegion(i).meanStdMask.RGB =
   meanStdRoofMaskR & meanStdRoofMaskG & meanStdRoofMaskB;

IS.roofTypes(idx).colorRegion(i).meanStdMask.H =
   meanStdRoofMaskH;
IS.roofTypes(idx).colorRegion(i).meanStdMask.S =
   meanStdRoofMaskS;
IS.roofTypes(idx).colorRegion(i).meanStdMask.V =
   meanStdRoofMaskV;
IS.roofTypes(idx).colorRegion(i).meanStdMask.HSV =
   meanStdRoofMaskH & meanStdRoofMaskS & meanStdRoofMaskV;
```

```
            IS.roofTypes(idx).colorRegion(i).meanStdMask.total = IS.
               roofTypes(idx).colorRegion(i).meanStdMask.RGB & ...
                IS.roofTypes(idx).colorRegion(i).meanStdMask.HSV;

        end

        meanStdMaskArray = [IS.roofTypes(idx).colorRegion.meanStdMask];
        minMaxMaskArray = [IS.roofTypes(idx).colorRegion.minMaxMask];

        IS.roofTypes(idx).mask = struct();
        IS.roofTypes(idx).mask.firstMask = struct();
        IS.roofTypes(idx).mask.firstMask.meanStd = max(cat(3,
           meanStdMaskArray.total),[],3);
        IS.roofTypes(idx).mask.firstMask.minMax = max(cat(3,
           minMaxMaskArray.total),[],3);

    end

    function IS = filterMask(IS,idx)

        nErosion = 2;
        nDilation = 2;

        IS.roofTypes(idx).mask.openClosedMask = struct();

        IS.roofTypes(idx).mask.openClosedMask.openedMask = struct();
        IS.roofTypes(idx).mask.openClosedMask.openedMask.minMax =
           myOpen(IS.roofTypes.mask.firstMask.minMax, [0 1 0; 1 1 1; 0
           1 0],nErosion,nDilation);
        IS.roofTypes(idx).mask.openClosedMask.openedMask.meanStd =
           myOpen(IS.roofTypes.mask.firstMask.meanStd, [0 1 0; 1 1 1; 0
            1 0],nErosion,nDilation);

        IS.roofTypes(idx).mask.openClosedMask.closedMask = struct();
        IS.roofTypes(idx).mask.openClosedMask.closedMask.minMax =
           myClose(IS.roofTypes.mask.firstMask.minMax, [0 1 0; 1 1 1; 0
            1 0],nErosion,nDilation);
        IS.roofTypes(idx).mask.openClosedMask.closedMask.meanStd =
           myClose(IS.roofTypes.mask.firstMask.meanStd, [0 1 0; 1 1 1;
           0 1 0],nErosion,nDilation);

        IS.roofTypes(idx).mask.openClosedMask.filledMask = struct();
        IS.roofTypes(idx).mask.openClosedMask.filledMask.minMax =
           imfill(IS.roofTypes.mask.firstMask.minMax,"holes");
```

```
    IS.roofTypes(idx).mask.openClosedMask.filledMask.meanStd =
       imfill(IS.roofTypes.mask.firstMask.meanStd,"holes");

    IS.roofTypes(idx).mask.openClosedMask.openedFilledMask = struct
       ();
    IS.roofTypes(idx).mask.openClosedMask.openedFilledMask.minMax =
        imfill(IS.roofTypes(idx).mask.openClosedMask.openedMask.
       minMax,"holes");
    IS.roofTypes(idx).mask.openClosedMask.openedFilledMask.meanStd
       = imfill(IS.roofTypes(idx).mask.openClosedMask.openedMask.
       meanStd,"holes");

    IS.roofTypes(idx).mask.openClosedMask.closedFilledMask = struct
       ();
    IS.roofTypes(idx).mask.openClosedMask.closedFilledMask.minMax =
        imfill(IS.roofTypes(idx).mask.openClosedMask.closedMask.
       minMax,"holes");
    IS.roofTypes(idx).mask.openClosedMask.closedFilledMask.meanStd
       = imfill(IS.roofTypes(idx).mask.openClosedMask.closedMask.
       meanStd,"holes");

    IS.roofTypes(idx).mask.openClosedMask.filledOpenedMask = struct
       ();
    IS.roofTypes(idx).mask.openClosedMask.filledOpenedMask.minMax =
        myOpen(IS.roofTypes(idx).mask.openClosedMask.filledMask.
       minMax, [0 1 0; 1 1 1; 0 1 0],nErosion,nDilation);
    IS.roofTypes(idx).mask.openClosedMask.filledOpenedMask.meanStd
       = myOpen(IS.roofTypes(idx).mask.openClosedMask.filledMask.
       meanStd, [0 1 0; 1 1 1; 0 1 0],nErosion,nDilation);

    IS.roofTypes(idx).mask.openClosedMask.filledClosedMask = struct
       ();
    IS.roofTypes(idx).mask.openClosedMask.filledClosedMask.minMax =
        myClose(IS.roofTypes(idx).mask.openClosedMask.filledMask.
       minMax, [0 1 0; 1 1 1; 0 1 0],nErosion,nDilation);
    IS.roofTypes(idx).mask.openClosedMask.filledClosedMask.meanStd
       = myClose(IS.roofTypes(idx).mask.openClosedMask.filledMask.
       meanStd, [0 1 0; 1 1 1; 0 1 0],nErosion,nDilation);

end

function openedIm = myOpen(ID,SE,nErosion,nDilation)

    openedIm = ID;
```

```
    for i = 1:nErosion
        openedIm = imerode(openedIm,SE);
    end
    for i = 1:nDilation
        openedIm = imdilate(openedIm,SE);
    end
end

function openedIm = myClose(ID,SE,nErosion,nDilation)

    openedIm = ID;

    for i = 1:nDilation
        openedIm = imdilate(openedIm,SE);
    end
    for i = 1:nErosion
        openedIm = imerode(openedIm,SE);
    end
end

function filledFilteredMask = filterIndividualMask(initialMask,
   minSize1, minSize2)

    filledMask = myFill(initialMask,400);
    openedFilledMask = imopen(filledMask,[0 1 0 ; 1 1 1 ; 0 1 0]);
    figure("Name","Opened filled mask");
    imshow(filledMask);

    % Suppress smaller blobs
    % Extract blobs and blob area
    imLabels = bwlabel(openedFilledMask);
    props = regionprops(openedFilledMask,imLabels,'Area');

    % Surpress small blobs
    k = find([props.Area] < minSize1);
    suppressedLabels = ismember(imLabels,k);
    filteredMask = xor(openedFilledMask,suppressedLabels);
    figure("Name","First filtered mask");
    imshow(filteredMask);

    closedMask = myClose(filteredMask,[0 1 0 ; 1 1 1 ; 0 1 0],2,2);
    filledFilteredMask = myFill(closedMask,800);

    % Suppress smaller blobs again
    % Extract blobs and blob area
```

```
    imLabels = bwlabel(filledFilteredMask);
    props = regionprops(filledFilteredMask,imLabels,'Area');

    % Surpress small blobs
    k = find([props.Area] < minSize2);
    suppressedLabels = ismember(imLabels,k);
    filteredMask = xor(filledFilteredMask,suppressedLabels);
    figure("Name","Twice filtered mask");
    imshow(filteredMask);

    % Finally close the image
    closedMask = myClose(filteredMask,[0 1 0 ; 1 1 1 ; 0 1 0],4,4);
    filledFilteredMask = myFill(closedMask,1000);
    figure("Name","Final filled closed filtered mask");
    imshow(filledFilteredMask);

end

function filledFilteredMask = filterIndividualGreyMask(initialMask,
    minSize1, minSize2)

    filledMask = myFill(initialMask,400);
    openedFilledMask = myOpen(filledMask,[0 1 0 ; 1 1 1 ; 0 1
       0],2,2);
    figure("Name","Opened filled mask");
    imshow(filledMask);

    % Suppress smaller blobs
    % Extract blobs and blob area
    imLabels = bwlabel(openedFilledMask);
    props = regionprops(openedFilledMask,imLabels,'Area');

    % Surpress small blobs
    k = find([props.Area] < minSize1);
    suppressedLabels = ismember(imLabels,k);
    filteredMask = xor(openedFilledMask,suppressedLabels);
    figure("Name","First filtered mask");
    imshow(filteredMask);

    closedMask = myClose(filteredMask,[0 1 0 ; 1 1 1 ; 0 1 0],2,2);
    filledFilteredMask = myFill(closedMask,700);

    % Suppress smaller blobs again
    % Extract blobs and blob area
    imLabels = bwlabel(filledFilteredMask);
```

```
    props = regionprops(filledFilteredMask,imLabels,'Area');

    % Surpress small blobs
    k = find([props.Area] < minSize2);
    suppressedLabels = ismember(imLabels,k);
    filteredMask = xor(filledFilteredMask,suppressedLabels);
    figure("Name","Twice filtered mask");
    imshow(filteredMask);

    % Finally close the image
    closedMask = myClose(filteredMask,[0 1 0 ; 1 1 1 ; 0 1 0],4,4);
    filledFilteredMask = myFill(closedMask,1000);
    figure("Name","Final filled closed filtered mask");
    imshow(filledFilteredMask);

end

function orMask = blobOR(mask1,mask2)

    mask1Labels = bwlabel(mask1);
    mask2Labels = bwlabel(mask2);
    blob1Values = unique(mask1Labels(mask2));
    blob2Values = unique(mask2Labels(mask1));
    orMask = ismember(mask1Labels,blob1Values(blob1Values>0))|
       ismember(mask2Labels,blob2Values(blob2Values>0));

end

function croppedBlobs = cropBlobs(blobMask)

f = figure('Name','Selected FAs for elimination');
imshow(blobMask);

key = 0;
selectedBlobs = zeros(size(blobMask));
% While enter is not pressed
while key ~= 13

    % Wait until some key or button is pressed and get the position
        of the
    % mouse and the pressed key.
    [x,y,key] = ginput(1);

    % If it was not enter
    if key ~= 13
```

```
        % If it was the left click add that FA to the deletion list
            .
        if key == 1
            selectedBlobs = selectedBlobs | bwselect(blobMask,x,y
               ,4);
            % If it was the right click remove the FA from the
               deletion
            % list.
        elseif key == 3
            selectedBlobs = xor(selectedBlobs,bwselect(blobMask,x,y
               ,4));
        end
    end

    % Refresh the image and show the selected FAs in red.
    selectedBlobsRGB = cat(3,blobMask,blobMask.*(1-selectedBlobs),
       blobMask.*(1-selectedBlobs));
    figure(f);
    imshow(selectedBlobsRGB,[]);
end

croppedBlobs = logical(blobMask - selectedBlobs);
close(f)

end

function visualizeMask(img,mask)
    figure("Name","Mask visualisation")
    imshow(img);
    hold on
    mask3D = cat(3,mask*0.1,mask*0.6,mask*0.7);
    I = imshow(mask3D);
    I.AlphaData = mask*0.6;
end

function finalMask = myFill(mask,maxArea)
    filledMask = imfill(mask,"holes");
    holeMask = xor(mask,filledMask);

    labelledHoles = bwlabel(holeMask);
    props = regionprops(holeMask,labelledHoles,'Area');

    k = find([props.Area] < maxArea);
    holes2fill = ismember(labelledHoles,k);
```

```
finalMask = mask|holes2fill;
```

Bibliography

Abass, F et al. (Jan. 2020). ‘A Review of Green Roof: Definition, History, Evolution and Functions’. In: *IOP Conference Series: Materials Science and Engineering* 713.1, p. 012048. DOI: `10.1088/1757-899X/713/1/012048` (cit. on p. 10).

Agra, Har’el et al. (2017). ‘Measuring the effect of plant-community composition on carbon fixation on green roofs’. In: *Urban Forestry & Urban Greening* 24, pp. 1–4. ISSN: 1618-8667. DOI: `https://doi.org/10.1016/j.ufug.2017.03.003` (cit. on p. 10).

Akbari, Hashem, Ronnen Levinson and Leo Rainer (2005). ‘Monitoring the energy-use effects of cool roofs on California commercial buildings’. In: *Energy and Buildings* 37.10, pp. 1007–1016. ISSN: 0378-7788. DOI: `https://doi.org/10.1016/j.enbuild.2004.11.013` (cit. on p. 9).

AndrÃ©s-DomÃ©nech, Ignacio et al. (2018). ‘Hydrological Performance of Green Roofs at Building and City Scales under Mediterranean Conditions’. In: *Sustainability* 10.9. ISSN: 2071-1050. DOI: `10.3390/su10093105` (cit. on p. 10).

Anuario estadístico de Valencia 2022 (2022). Tech. rep. (cit. on p. 46).

Benedito DurÃ, Vicent et al. (2023). ‘Contribution of green roofs to urban arthropod biodiversity in a Mediterranean climate: A case study in ValÃ¨ncia, Spain’. In: *Building and Environment* 228, p. 109865. ISSN: 0360-1323. DOI: `https://doi.org/10.1016/j.buildenv.2022.109865` (cit. on p. 10).

Bianchini, Fabricio and Kasun Hewage (2012). ‘How “green” are the green roofs? Lifecycle analysis of green roof materials’. In: *Building and Environment* 48, pp. 57–65. ISSN: 0360-1323. DOI: `https://doi.org/10.1016/j.buildenv.2011.08.019` (cit. on p. 12).

Borràs, Júlia G. et al. (2022). ‘Energy Efficiency Evaluation of Green Roofs as an Intervention Strategy in Residential Buildings in the Field of Spanish Climate’. In: *Buildings* 12.7. ISSN: 2075-5309. DOI: `10.3390/buildings12070959` (cit. on pp. 10, 28).

Cao, Xiaodong, Xilei Dai and Junjie Liu (2016). 'Building energy-consumption status worldwide and the state-of-the-art technologies for zero-energy buildings during the past decade'. In: *Energy and Buildings* 128, pp. 198–213. ISSN: 0378-7788. DOI: `https://doi.org/10.1016/j.enbuild.2016.06.089` (cit. on p. 9).

Castellar, J.A.C. et al. (2021). 'Nature-based solutions in the urban context: terminology, classification and scoring for urban challenges and ecosystem services'. In: *Science of The Total Environment* 779, p. 146237. ISSN: 0048-9697. DOI: `https://doi.org/10.1016/j.scitotenv.2021.146237` (cit. on pp. 8–10).

Castleton, H.F. et al. (2010). 'Green roofs; building energy savings and the potential for retrofit'. In: *Energy and Buildings* 42.10, pp. 1582–1591. ISSN: 0378-7788. DOI: `https://doi.org/10.1016/j.enbuild.2010.05.004` (cit. on p. 10).

Centro de Descargas (2023). URL: `https://centrodedescargas.cnig.es/CentroDescargas/index.jsp` (visited on 15/07/2023) (cit. on pp. 17, 32).

Chen, Lin et al. (Aug. 2022). 'Strategies to achieve a carbon neutral society: a review'. In: *Environmental Chemistry Letters* 20.4, pp. 2277–2310. ISSN: 1610-3661. DOI: `10.1007/s10311-022-01435-8` (cit. on p. 7).

Ciampi, M., F. Leccese and G. Tuoni (2003). 'Ventilated facades energy performance in summer cooling of buildings'. In: *Solar Energy* 75.6, pp. 491–502. ISSN: 0038-092X. DOI: `https://doi.org/10.1016/j.solener.2003.09.010` (cit. on p. 9).

Código Técnico Estructural (CTE) - Documento básico HS - Salubridad (June 2022). Standard. Madrid, Spain (cit. on p. 11).

Conference "The UPV-City alliance for the Climate Mission 2030" (Oct. 2022). URL: `https://www.upv.es/entidades/vcampus/en/2022/10/24/conference-the-upv-city-alliance-for-the-climate-mission-2030/` (visited on 09/01/2023) (cit. on p. 4).

Datos Energéticos de la Comunitat Valenciana (2020) (2023). Tech. rep. (cit. on p. 36).

Departamento de Medio Ambiente, Planificación Territorial y Vivienda del País Vasco (2016). *Soluciones Naturales para la adaptación al cambio climático en el ámbito local de la Comunidad Autónoma del País Vasco.* Tech. rep. (cit. on pp. 8, 9).

Donati, Giulia F.A. et al. (2022). 'Reconciling cities with nature: Identifying local Blue-Green Infrastructure interventions for regional biodiversity enhancement'. In: *Journal of Environmental Management* 316, p. 115254. ISSN: 0301-4797. DOI: `https://doi.org/10.1016/j.jenvman.2022.115254` (cit. on p. 10).

Ekmekcioğlu, Ömer (2023). 'On the identification of most appropriate green roof types for urbanized cities using multi-tier decision analysis: A case study of Istanbul, Turkey'. In: *Sustainable Cities and Society* 96, p. 104707. ISSN: 2210-6707. DOI: `https://doi.org/10.1016/j.scs.2023.104707` (cit. on p. 12).

Energy performance of buildings directive (2023). URL: `https://energy.ec.europa.eu/topics/energy-efficiency/energy-efficient-buildings/energy-performance-buildings-directive_en#facts-and-figures` (visited on 09/06/2023) (cit. on p. 8).

EU Mission: Climate-Neutral and Smart Cities (2023). URL: `https://research-and-innovation.ec.europa.eu/funding/funding-opportunities/funding-programmes-and-open-calls/horizon-europe/eu-missions-horizon-europe/climate-neutral-and-smart-cities_en` (visited on 09/06/2023) (cit. on p. 3).

EU Missions in Horizon Europe (2023). URL: `https://research-and-innovation.ec.europa.eu/funding/funding-opportunities/funding-programmes-and-open-calls/horizon-europe/eu-missions-horizon-europe_en` (visited on 09/06/2023) (cit. on p. 3).

European Green Deal (July 2023). URL: `https://www.consilium.europa.eu/en/policies/green-deal/` (visited on 09/06/2023) (cit. on p. 3).

European Parliment and Council of European Union (2018). *Consolidated text: Directive 2010/31/EU of the European Parliament and of the Council of 19 May 2010 on the energy performance of buildings (recast).*
`http://data.europa.eu/eli/dir/2010/31/2021-01-01` (cit. on p. 8).

Factores de emisión. Registro de huella de carbono, compensación y proyectos de absorción de dióxido de carbono. (2023). Tech. rep. (cit. on p. 36).

Fantozzi, Fabio et al. (2021). 'Do green roofs really provide significant energy saving in a Mediterranean climate? Critical evaluation based on different case studies'. In: *Frontiers of Architectural Research* 10.2, pp. 447–465. ISSN: 2095-2635. DOI: `https://doi.org/10.1016/j.foar.2021.01.006` (cit. on p. 10).

Foustalieraki, M. et al. (2017). 'Energy performance of a medium scale green roof system installed on a commercial building using numerical and experimental data recorded during the cold period of the year'. In: *Energy and Buildings* 135, pp. 33–38. ISSN: 0378-7788. DOI: `https://doi.org/10.1016/j.enbuild.2016.10.056` (cit. on p. 10).

García-Prieto, Alejandra, Begoña Serrano and Leticia Ortega (2016). *Catálogo de tipología edificatoria residencial. Ámbito: España.* Tech. rep. (cit. on p. 35).

Generador de precios de la construcción. (N.d.). URL: `http://www.generadordeprecios.info/#gsc.tab=0` (cit. on p. 47).

Gong, Yongwei et al. (2020). 'Factors affecting the ability of extensive green roofs to reduce nutrient pollutants in rainfall runoff'. In: *Science of The Total Environment* 732, p. 139248. ISSN: 0048-9697. DOI: `https://doi.org/10.1016/j.scitotenv.2020.139248` (cit. on p. 10).

Green Roof Guidelines - Guidelines for the Planning, Construction and Maintenance of Green Roofs (2018). Tech. rep. (cit. on p. 11).

Heidarinejad, Ghassem and Arash Esmaili (2015). 'Numerical simulation of the dual effect of green roof thermal performance'. In: *Energy Conversion and Management* 106, pp. 1418–1425. ISSN: 0196-8904. DOI: `https://doi.org/10.1016/j.enconman.2015.10.020` (cit. on p. 10).

Horizon Europe (2023). URL: `https://research-and-innovation.ec.europa.eu/funding/funding-opportunities/funding-programmes-and-open-calls/horizon-europe_en` (visited on 09/06/2023) (cit. on p. 3).

Jaffal, Issa, Salah-Eddine Ouldboukhitine and Rafik Belarbi (2012). 'A comprehensive study of the impact of green roofs on building energy performance'. In: *Renewable Energy* 43, pp. 157–164. ISSN: 0960-1481. DOI: `https://doi.org/10.1016/j.renene.2011.12.004` (cit. on p. 10).

Kuznik, FrÃ©dÃ©ric and Joseph Virgone (2009). 'Experimental assessment of a phase change material for wall building use'. In: *Applied Energy* 86.10, pp. 2038–2046. ISSN: 0306-2619. DOI: `https://doi.org/10.1016/j.apenergy.2009.01.004` (cit. on p. 9).

Liu, Wen et al. (2019). 'The impacts of substrate and vegetation on stormwater runoff quality from extensive green roofs'. In: *Journal of Hydrology* 576, pp. 575–582. ISSN: 0022-1694. DOI: `https://doi.org/10.1016/j.jhydrol.2019.06.061` (cit. on p. 10).

LLC, Google (2023). *Google Earth Pro*. Version 7.3.6 (cit. on pp. 17, 19, 20, 22, 24–27, 38–40, 65–71).

Manzano-Agugliaro, Francisco et al. (2015). 'Review of bioclimatic architecture strategies for achieving thermal comfort'. In: *Renewable and Sustainable Energy Reviews* 49, pp. 736–755. ISSN: 1364-0321. DOI: `https://doi.org/10.1016/j.rser.2015.04.095` (cit. on p. 9).

Massaro, Emanuele et al. (May 2023). 'Spatially-optimized urban greening for reduction of population exposure to land surface temperature extremes'. In: *Nature Communications* 14.1, p. 2903. ISSN: 2041-1723. DOI: `10.1038/s41467-023-38596-1` (cit. on p. 10).

MathWorks (2022). *MATLAB*. Version R2022b (cit. on pp. 17, 24–27, 42–45, 65–71).

Md. Yacob, Mohamad Norfekry, Hartini Kasmin and Muhammad Iqbal Hakeem Hashim (June 2021). 'Estimating Carbon Sequestration of Green Roof Plants in Tropical Climate'. In: *International Journal of Integrated Engineering* 13.3, pp. 200–206 (cit. on p. 10).

Mihalakakou, Giouli et al. (2023). 'Green roofs as a nature-based solution for improving urban sustainability: Progress and perspectives'. In: *Renewable and Sustainable Energy Reviews* 180, p. 113306. ISSN: 1364-0321. DOI: `https://doi.org/10.1016/j.rser.2023.113306` (cit. on p. 9).

Moran, D. et al. (2022). 'Estimating CO_2 emissions for 108 000 European cities'. In: *Earth System Science Data* 14.2, pp. 845–864. DOI: `10.5194/essd-14-845-2022` (cit. on p. 7).

Nature-based solutions (2023). URL: `https://research-and-innovation.ec.europa.eu/research-area/environment/nature-based-solutions_en` (visited on 09/06/2023) (cit. on p. 8).

Nature-based solutions research policy (2023). URL: `https://research-and-innovation.ec.europa.eu/research-area/environment/nature-based-solutions/research-policy_en` (visited on 09/06/2023) (cit. on p. 8).

Normas tecnológicas de Jardinería y Paisajismo - 11C - Ajardinamientos especiales: Cubiertas verdes (2012). Tech. rep. (cit. on pp. 10, 11).

Oberndorfer, Erica et al. (Nov. 2007). 'Green Roofs as Urban Ecosystems: Ecological Structures, Functions, and Services'. In: *BioScience* 57.10, pp. 823–833. ISSN: 0006-3568. DOI: `10.1641/B571005`. eprint: `https://academic.oup.com/bioscience/article-pdf/57/10/823/27021976/57-10-823.pdf` (cit. on p. 10).

Office of the Chief Building Official, Toronto Building (2010). *Toronto Green Roof Construction Standard. Supplementary Guidelines.* Tech. rep. (cit. on p. 11).

Oficina de estadística - Barrios 2023 (2022). URL: `https://www.valencia.es/es/cas/estadistica/mapa-barrios` (visited on 16/08/2023) (cit. on p. 31).

Ondoño, S., J.J. Martínez-Sánchez and J.L. Moreno (2016). 'The inorganic component of green roof substrates impacts the growth of Mediterranean plant species as well as the C and N sequestration potential'. In: *Ecological Indicators* 61, pp. 739–752. ISSN: 1470-160X. DOI: `https://doi.org/10.1016/j.ecolind.2015.10.025` (cit. on pp. 10, 27).

Open Source Geospatial Foundation (2020). *QGIS*. Version 3.16.16 (cit. on pp. 17–19, 21, 33, 34, 42).

Pan, Haozhi et al. (Aug. 2023). 'Contribution of prioritized urban nature-based solutions allocation to carbon neutrality'. In: *Nature Climate Change* 13.8, pp. 862–870. ISSN: 1758-6798. DOI: `10.1038/s41558-023-01737-x` (cit. on pp. 7, 10).

Peñalvo-López, Elisa et al. (2020). 'Study of the Improvement on Energy Efficiency for a Building in the Mediterranean Area by the Installation of a Green Roof System'. In: *Energies* 13.5. ISSN: 1996-1073. DOI: `10.3390/en13051246` (cit. on pp. 10, 28).

Portal de datos abiertos de Valencia (2023). URL: `https://valencia.opendatasoft.com/pages/home/` (visited on 07/07/2023) (cit. on p. 16).

Ranganathan, Janet et al. (Jan. 2004). *WBCSD/WRI, 2004. Greenhouse Gas Protocol: a Corporate Accounting and Reporting Standard.* DOI: `10.13140/RG.2.2.34895.33443` (cit. on p. 7).

Rosenzweig, C., S. Gaffin and L. Parshall, eds. (2006). *Green Roofs in the New York Metropolitan Region: Research Report.* New York, N.Y. (cit. on p. 10).

Sadineni, Suresh B., Srikanth Madala and Robert F. Boehm (2011). 'Passive building energy savings: A review of building envelope components'. In: *Renewable and Sustainable Energy Reviews* 15.8, pp. 3617–3631. ISSN: 1364-0321. DOI: `https://doi.org/10.1016/j.rser.2011.07.014` (cit. on p. 9).

Saiz, Susana et al. (2006). 'Comparative Life Cycle Assessment of Standard and Green Roofs'. In: *Environmental Science & Technology* 40.13. PMID: 16856752, pp. 4312–4316. DOI: `10.1021/es0517522`. eprint: `https://doi.org/10.1021/es0517522` (cit. on pp. 10–12, 28).

Santamouris, M. (2014). 'Cooling the cities - A review of reflective and green roof mitigation technologies to fight heat island and improve comfort in urban environments'. In: *Solar Energy* 103, pp. 682–703. ISSN: 0038-092X. DOI: `https://doi.org/10.1016/j.solener.2012.07.003` (cit. on p. 10).

Santos, Teresa, José António Tenedório and José Alberto Gonçalves (2016). 'Quantifying the City's Green Area Potential Gain Using Remote Sensing Data'. In: *Sustainability* 8.12. ISSN: 2071-1050. DOI: `10.3390/su8121247` (cit. on pp. 13, 18).

Schindelin, Johannes et al. (July 2012). 'Fiji: an open-source platform for biological-image analysis'. In: *Nature Methods* 9.7, pp. 676–682. ISSN: 1548-7105. DOI: `10.1038/nmeth.2019` (cit. on p. 22).

Servicios INSPIRE de Cartografía Catastral – Edificios (2023). URL: `https://www.catastro.minhap.es/webinspire/index.html` (visited on 07/07/2023) (cit. on p. 16).

Shafique, Muhammad, Reeho Kim and Muhammad Rafiq (2018). 'Green roof benefits, opportunities and challenges - A review'. In: *Renewable and Sustainable Energy Reviews* 90, pp. 757–773. ISSN: 1364-0321. DOI: `https://doi.org/10.1016/j.rser.2018.04.006` (cit. on p. 10).

Shafique, Muhammad, Xiaolong Xue and Xiaowei Luo (2020). 'An overview of carbon sequestration of green roofs in urban areas'. In: *Urban Forestry & Urban Greening* 47, p. 126515. ISSN: 1618-8667. DOI: `https://doi.org/10.1016/j.ufug.2019.126515` (cit. on pp. 10, 27).

Shao, Huamei et al. (2021). 'Assessing city-scale green roof development potential using Unmanned Aerial Vehicle (UAV) imagery'. In: *Urban Forestry & Urban Greening* 57, p. 126954. ISSN: 1618-8667. DOI: `https://doi.org/10.1016/j.ufug.2020.126954` (cit. on p. 13).

Sibaruddin, H I et al. (June 2018). 'Comparison of pixel-based and object-based image classification techniques in extracting information from UAV imagery data'. In: *IOP Conference Series: Earth and Environmental Science* 169.1, p. 012098. DOI: `10.1088/1755-1315/169/1/012098` (cit. on p. 32).

Silva, Cristina Matos, Inês Flores-Colen and Maria Antunes (2017). 'Step-by-step approach to ranking green roof retrofit potential in urban areas: A case study of Lisbon, Portugal'. In: *Urban Forestry & Urban Greening* 25, pp. 120–129. ISSN: 1618-8667. DOI: `https://doi.org/10.1016/j.ufug.2017.04.018` (cit. on pp. 11, 12, 21).

Slootweg, Mike et al. (2023). 'Identifying the geographical potential of rooftop systems: Space competition and synergy'. In: *Urban Forestry & Urban Greening* 79, p. 127816. ISSN: 1618-8667. DOI: `https://doi.org/10.1016/j.ufug.2022.127816` (cit. on p. 13).

The City Council and the Polytechnic University of Valencia join forces in order to promote the València 2030 Neutral City plan (2023). URL: `https://www.missionsvalencia.eu/lajuntament-i-la-universitat-politecnica-de-valencia-salien-per-impulsar-el-pla-de-valencia-ciutat-neutra-2030/?lang=en` (visited on 09/01/2023) (cit. on p. 4).

Timeline - European Green Deal and fit for 55 (2023). URL: `https://www.consilium.europa.eu/en/policies/green-deal/timeline-european-green-deal-and-fit-for-55/` (visited on 09/06/2023) (cit. on p. 3).

Turiel, Antonio (2020). 'Por qué no funcionan el ahorro y la eficiencia'. In: *Petrocalipsis*. Madrid: Alfabeto. Chap. 14 (cit. on p. 7).

Velasco, Erik et al. (2016). 'Does urban vegetation enhance carbon sequestration?' In: *Landscape and Urban Planning* 148, pp. 99–107. ISSN: 0169-2046. DOI: `https://doi.org/10.1016/j.landurbplan.2015.12.003` (cit. on p. 10).

Velázquez, J. et al. (2019). 'Planning and selection of green roofs in large urban areas. Application to Madrid metropolitan area'. In: *Urban Forestry & Urban Greening* 40. Urban green infrastructure - connecting people and nature for sustainable cities, pp. 323–334. ISSN: 1618-8667. DOI: `https://doi.org/10.1016/j.ufug.2018.06.020` (cit. on pp. 13, 34).

Wang, Jian et al. (Feb. 2023). 'Can low-carbon pilot policies improve the efficiency of urban carbon emissions? A quasi-natural experiment based on 282 prefecture-level cities across China'. In: *PLOS ONE* 18.2, pp. 1–21. DOI: `10.1371/journal.pone.0282109` (cit. on p. 7).

Williams, Kathryn J.H. et al. (2019). 'Appraising the psychological benefits of green roofs for city residents and workers'. In: *Urban Forestry & Urban Greening* 44, p. 126399. ISSN: 1618-8667. DOI: `https://doi.org/10.1016/j.ufug.2019.126399` (cit. on p. 10).

Yang, Jun, Qian Yu and Peng Gong (2008). 'Quantifying air pollution removal by green roofs in Chicago'. In: *Atmospheric Environment* 42.31, pp. 7266–7273. ISSN: 1352-2310. DOI: `https://doi.org/10.1016/j.atmosenv.2008.07.003` (cit. on p. 10).