Bachelor's thesis

BUSINESS EVALUATION OF ML SYSTEM FOR SATELLITE-BASED URBAN HOTSPOTS PREDICTIONS

Ivan Anikin

Faculty of Information Technology
Department of Applied Mathematics
Supervisor: Mgr. Alexander Kovalenko, Ph.D.
April 22, 2025

FACULTY OF INFORMATION TECHNOLOGY CTU IN PRAGUE

Assignment of bachelor's thesis

Title:	Business Evaluation of ML System for Satellite-Based Urban Hotspots Predictions
Student:	Ivan Anikin
Supervisor:	Mgr. Alexander Kovalenko, Ph.D.
Study program:	Informatics
Branch / specialization:	Business Informatics 2021
Department:	Department of Software Engineering
Validity:	until the end of summer semester 2025/2026

Instructions

Background:
Climate change is increasingly affecting urban environments, with extreme heat events posing significant challenges to city infrastructure, public health, and local economies. Decision-makers require actionable insights to mitigate these impacts, especially in resource-constrained settings. Leveraging historic Earth observation data from NASA and ESA satellites, this project aims to develop a machine learning-based forecasting tool to predict the economic and infrastructural effects of extreme heat events in urban areas. The thesis includes a comprehensive evaluation of the tool's economic feasibility and management implications, such as cost-benefit analysis, scalability for city planners, and utility for public health strategies. The resulting insights aim to empower municipalities and businesses to optimize resource allocation, prioritize interventions, and increase urban resilience.
1. Literature Review:
Examine research on machine learning applications for urban heat prediction and their economic or business impacts.
Explore case studies where predictive tools have been implemented to assess their cost-effectiveness and value for stakeholders, such as municipalities or private enterprises.
2. Economic Impact Assessment Framework:
Identify key economic indicators affected by urban heat events, such as healthcare costs, energy expenditures, and infrastructure maintenance.
Develop a framework to quantify the potential economic benefits of using a forecasting

---

Electronically approved by Ing. David Buchtela, Ph.D. on 23 January 2025 in Prague.

tool for heat mitigation.
3. Business Case Development:
Outline potential applications of the tool for different stakeholders, including city planners, public health agencies, and energy providers.
Estimate the return on investment (ROI) for implementing the tool, considering factors like reduced operational costs, improved planning efficiency, and enhanced public health outcomes.
4. Data Collection and Models Design:
Acquire and prepare satellite data, focusing on urban regions with significant economic and infrastructural vulnerabilities to heat events.
Investigate deep learning architectures suitable for large-scale spatiotemporal data.

5. Prototype Development:
Implement chosen models, defining the appropriate network architectures, layers, and hyperparameters.
Design a user-friendly prototype UI of the forecasting tool that includes visualizations of economic impacts, such as cost-benefit analyses, ROI estimates, and actionable insights for stakeholders.
Ensure the tool supports customizable scenarios to address specific urban planning or business needs.
6. Prototype Demonstration and Feedback:
Present the prototype to potential users, including city planners and business stakeholders, to gather feedback on usability, economic insights, and practical value.
Incorporate feedback into refining the tool, focusing on its adaptability to diverse urban and economic contexts.
7. Conclusion and Future Directions:
Summarize the tool's potential economic and business contributions to mitigating urban heat impacts.
Propose future enhancements, such as integrating broader climate-related risk assessments or expanding the tool’s applications to global markets.
Discuss how improved spatial and temporal pattern extraction from satellite imagery can improve resilience against extreme heat events

I would like to give my special thanks to the supervisor of my work Mgr. Alexander Kovalenko, Ph.D., giving me directions and unseful information, making this thesis better. I'm grateful to the CTU university and mainly the Faculty of information technology I had the luck to study at. I am grateful to the supervisor of our specialisation Ing. David Buchtela, Ph.D. for giving me broad background knowledge in economics and business evaluation at his classes. Last but not least I'd like to thank Ecoten Urban Comfort and my fellow colleagues working with me on satelite images processing for several years now.

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic final thesis. I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended. In accordance with Section 2373(2) of Act No. 89/2012 Coll., the Civil Code, as amended, I hereby grant a non-exclusive authorization (licence) to utilize this thesis, including all computer programs that are part of it or attached to it and all documentation thereof (hereinafter collectively referred to as the "Work"), to any and all persons who wish to use the Work. Such persons are entitled to use the Work in any manner that does not diminish the value of the Work and for any purpose (including use for profit). This authorisation is unlimited in time, territory and quantity.

In Praze on April 22, 2025

Abstract

This thesis focuses on the use of machine learning and satellite data (Landsat, Sentinel) for predicting thermal maps in urban environments. Deep learning models, particularly CNN and U-Net architectures, were designed and tested to predict temperature fields based on environmental indicators. Over 200 models were trained and evaluated using different parameter combinations across various cities. The thesis includes an economic assessment of the predictive models' benefits for urban planning, along with a business plan and SWOT analysis.

Keywords machine learning, satellite imagery, climate forecasting, economic impact, cost-benefit analysis, remote sensing, urban planning, sustainable cities, deep learning, thermal data

Abstrakt

Tato práce se věnuje využití strojového učení a družicových dat (Landsat, Sentinel) pro predikci tepelných map ve městském prostředí. Byly navrženy a otestovány modely hlubokého učení, zejména architektury CNN a U-Net, pro predikci teplotních polí na základě environmentálních ukazatelů. V rámci této práce bylo natrénováno a vyhodnoceno přes 200 modelů s různými kombinacemi parametrů napříč různými městy. Součástí práce je ekonomické zhodnocení přínosu predikcí pro městské plánování a podnikatelksý záměr včetně SWOT analýzy.

Klíčová slova strojové učení, tepelné snímky, satelitní snímky, předpověď klimatu, ekonomický dopad, dálkový průzkum Země, udržitelná města

Contents

List of Figures

List of Tables

List of code listings

List of abbreviations

ML	Machine Learning
NN	Neural network
AI	Artificial Intelligence
CNN	Convolutional Neural Network
U-NET	Type of Convolutional Neural Network
LSTM	Long Short-Term Memory
LST	Land Surface Temperature
RNN	Recurrent neural network
CBA	Cost Benefit Analysis
UHI	Urban Heat Island
GIS	Geographical Information Systems
EO	Earth Observation
ROI	Return on Investment
MAE	Mean Absolute Error
MSE	Mean Squared Error
MI	Mutual Information
NDVI	Normalized Difference Vegetation Index
NDWI	Normalized Difference Water Index
EVI	Enhanced Vegetation Index
ESG	Environmental, Social, and Governance
IOT	Internet of Things

Chapter 1

Introduction

Extreme heat events pose a growing challenge to urban environments, affecting public health, infrastructure durability, and economic productivity. As cities expand, they replace natural land cover with dense concentrations of pavement, buildings, and other surfaces that absorb and retain heat. This leads to significant temperature differences between urban areas and their rural surroundings, particularly during the evening hours. The phenomenon not only increases energy demands for cooling but also escalates the risk of heat-related illnesses and mortality.

Leveraging NASA's Landsat and ESA's Sentinel datasets, this research develops an ML-based forecasting tool using convolutional neural networks (CNNs), long short-term memory (LSTM) networks, and remote sensing techniques to predict extreme heat events. The integration of geospatial analysis with deep learning enhances real-time climate monitoring, enabling data-driven urban planning and resource allocation.

Moreover, the availability of high-resolution Earth Observation (EO) data from satellite missions such as NASA's Landsat and the European Space Agency's Sentinel has revolutionized the monitoring and analysis of environmental changes associated with urban heat islands. By applying machine learning techniques to these data sets, it is possible to predict heat event occurrences with greater accuracy and to model potential future scenarios under different urban planning strategies.

The economic implications of UHI are profound, influencing public health, energy consumption, real estate, and urban sustainability. The strategic application of ML models in forecasting UHI effects can thus serve as a critical tool for urban planners and policy makers, helping to guide interventions that aim to mitigate these impacts. This thesis not only investigates the technical feasibility of such models but also evaluates their economic benefits and scalability, providing a comprehensive view of their practical applications. As the climate continues to change, the urgency of addressing urban heat issues becomes increasingly critical.

1.1 Objectives

The primary objective of this thesis is to develop a machine learning-based framework for forecasting urban heat islands (UHIs) using satellite imagery, and to evaluate its practical value from both economic and business perspectives. The work is structured around the following key goals:

- **Data Collection:** Acquire and preprocess high-resolution satellite data from sources such as NASA's Landsat and ESA's Sentinel missions, with a focus on thermal and vegetation-related indicators.
- **Model Development:** Design and implement a deep learning model, utilizing architectures such as convolutional neural networks (CNNs) and U-Net, to predict land surface temperature maps based on environmental features.
- **Economic Evaluation:** Quantify the potential cost savings in areas such as energy consumption, infrastructure maintenance, and healthcare by applying the prediction model to urban planning scenarios.
- **Business Case Analysis:** Explore the real-world applicability, scalability, and return on investment (ROI) of the solution for stakeholders including city planners, energy providers, and public health agencies. Demonstrate the model prototype to potential customers and parners.

Chapter 2
Background

"The urban heat island effect imposes real but often hidden costs on urban societies by reducing the efficiency and lifespan of cooling systems, increasing energy consumption, and leading to accelerated device failures." [1]

Numerous studies have demonstrated the effectiveness of ML for analyzing climate change phenomena. For example, convolutional neural networks (CNNs) have been extensively utilized for classifying land cover types, detecting urban sprawl, and analyzing vegetation changes. Recurrent neural networks (RNNs) and long short-term memory (LSTM) networks have shown promise in capturing temporal trends, such as seasonal variations and long-term climatic shifts. These methods enable a detailed understanding of how climate factors evolve over time, providing a foundation for predictive modeling.

2.1 Motivation

Urbanization has significantly altered natural landscapes, leading to complex environmental challenges such as the Urban Heat Island (UHI) effect. [2] With rising global temperatures, cities are experiencing more frequent and intense heatwaves, which adversely affect public health, infrastructure, and economic productivity. [3] Predicting these extreme events using machine learning (ML) models applied to satellite imagery presents an opportunity to enhance urban resilience and optimize resource allocation. [4] The availability of high-resolution Earth Observation (EO) data from NASA's Landsat and ESA's Sentinel missions has revolutionized climate modeling and forecasting. [5] Advances in deep learning and remote sensing analytics enable precise temperature predictions and early warning systems for urban planners. However, the economic feasibility and scalability of such predictive systems remain underexplored, making it essential to evaluate their real-world business applications.

2.2 Satelite Data

Satellite data play a crucial role in modern climate studies, offering high-resolution and multi-temporal observations of the Earth's surface. These datasets are collected through remote sensing technology, where satellites equipped with various sensors capture electromagnetic radiation reflected or emitted by land, water, and the atmosphere. The data collected provide insights into key environmental variables, such as land surface temperature (LST), vegetation indices, and urban expansion patterns, making them essential for urban heat island (UHI) analysis.

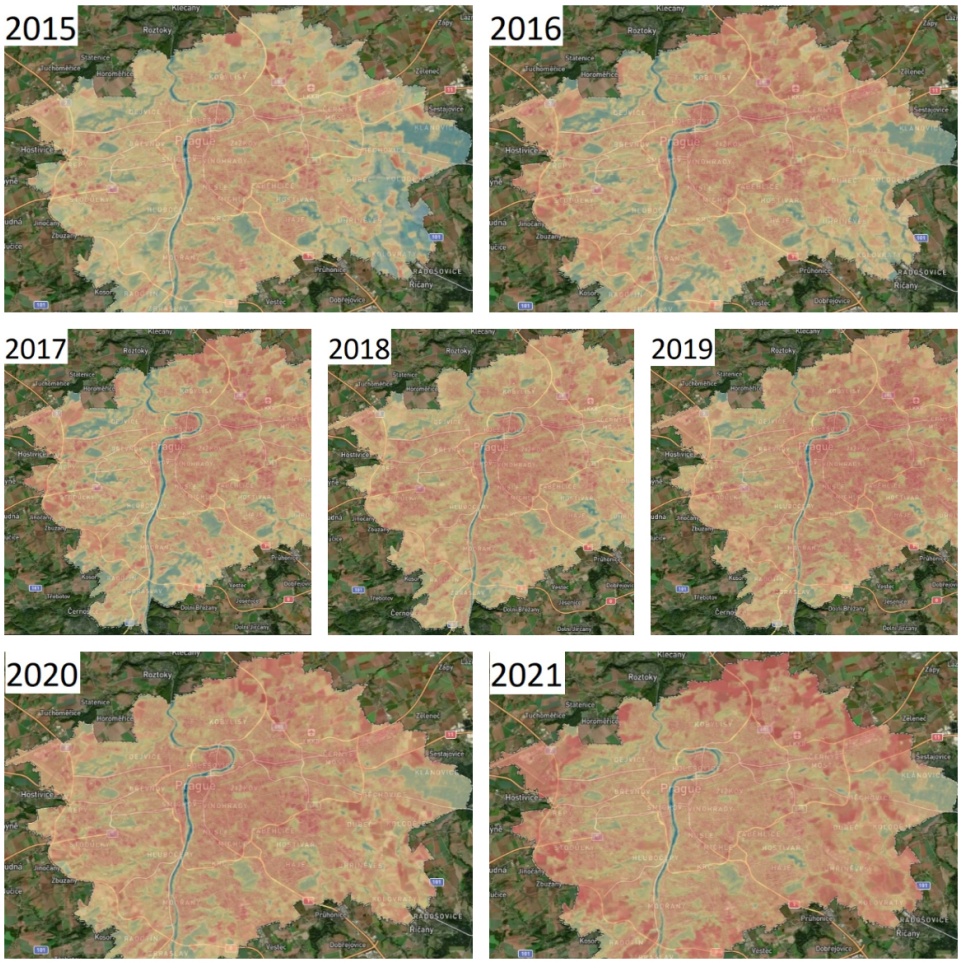

Figure 2.1 Prague Heat

The accompanying collage 2.1 of heat map developments in Prague demonstrates the use of satellite thermal imagery in studying urban heat islands (UHI). These maps highlight temperature variations across urban and suburban areas.

2.2.1 Types of Satellite Imagery

Satellite images are broadly categorized based on their spectral, spatial, and temporal resolutions. The primary types include:

- Optical Imagery: Captured in the visible and near-infrared spectrum, this type of data is widely used for land cover classification, vegetation monitoring, and urban development analysis. Notable sources include NASA's Landsat and ESA's Sentinel-2 missions.
- Thermal Imagery: Measures surface temperatures by detecting infrared radiation, making it crucial for UHI studies. Landsat-8 Thermal Infrared Sensor (TIRS) and Sentinel-3 Sea and Land Surface Temperature Radiometer (SLSTR) provide valuable thermal datasets for heat mapping.
- Radar and LiDAR Data: Active sensing technologies, such as synthetic aperture radar (SAR) from Sentinel-1 and LiDAR from airborne systems, are used for three-dimensional mapping of urban structures and terrain.
- Hyperspectral Imagery: Captures hundreds of spectral bands for detailed material composition analysis, aiding in distinguishing between vegetation types and built-up areas.

2.3 Machine Learning on Satellite Data

Recent developments in computer vision and artificial intelligence (AI) have enhanced the ability to analyze multi-spectral and thermal satellite imagery for urban heat mapping. [5] Machine learning models, particularly convolutional neural networks (CNNs) [6] and recurrent neural networks (RNNs) [7], have proven effective in extracting spatial and temporal patterns from large-scale EO data. [5] Some of the most widely used ML architectures in satellite-based urban heat predictions include:

- Convolutional Neural Networks (CNNs): Used for detecting patterns in land surface temperature (LST), vegetation indices, and urban structures. CNNs efficiently process multi-spectral images to classify heat zones. [8]
- Long Short-Term Memory (LSTM) Networks: Ideal for capturing temporal trends, such as seasonal temperature fluctuations, by analyzing historical climate data. [9]
- Random Forest and Gradient Boosting Models: Applied for feature selection and hybrid modeling of satellite-derived indicators, such as land cover changes and urban expansion trends. [10]

The integration of ML with Geographical Information Systems (GIS) [11] and remote sensing techniques has enabled real-time monitoring of environmental parameters. These innovations facilitate precise urban heat forecasts, allowing municipalities to implement proactive heat mitigation policies.

2.4 Benchmark Results in Urban Heat Studies

Benchmark studies in urban heat modeling have underscored the significance of integrating diverse data sources. For instance, research incorporating LST, NDVI, and EVI values from satellite imagery has achieved high prediction accuracy in urban heat mapping. Studies have also explored the role of contextual data, such as population density and socioeconomic indicators, to refine forecasting models. [12]

2.5 Difference-in-Differences Framework

The DID framework isolates specific environmental impacts on vegetation or urban areas, separating them from confounding variables. Applied to study heat stress and dry-hot winds, this approach leverages vegetation indices (VIs) derived from satellite data to model hazard impacts. Its flexibility makes it applicable to urban heat stress modeling, providing clear quantification of climate-related effects. [13]

2.6 Enhanced Indices

Spectral indices, derived from multi-spectral satellite imagery, play a vital role in analyzing environmental changes, particularly in urban heat island (UHI) studies. These indices are calculated using reflectance values from specific satellite bands, enabling the detection of subtle variations in land cover, vegetation health, and surface temperatures.

Several indices widely used in UHI and climate studies:

- Land Surface Temperature (LST):

  Derived from thermal infrared (TIR) bands, LST provides direct measurements of surface heat, which are critical for urban heat mapping. Satellites like Landsat 8 (TIRS) and Sentinel-3 (SLSTR) offer thermal data suitable for LST extraction.2.2

- Normalized Difference Water Index (NDWI): Using green (G) and near-infrared (NIR) bands, NDWI helps map water bodies and monitor moisture content, which influence urban microclimates and cooling patterns.2.3

- Enhanced Vegetation Index (EVI): Building upon NDVI, EVI incorporates blue (B) band data to correct for atmospheric distortions and canopy

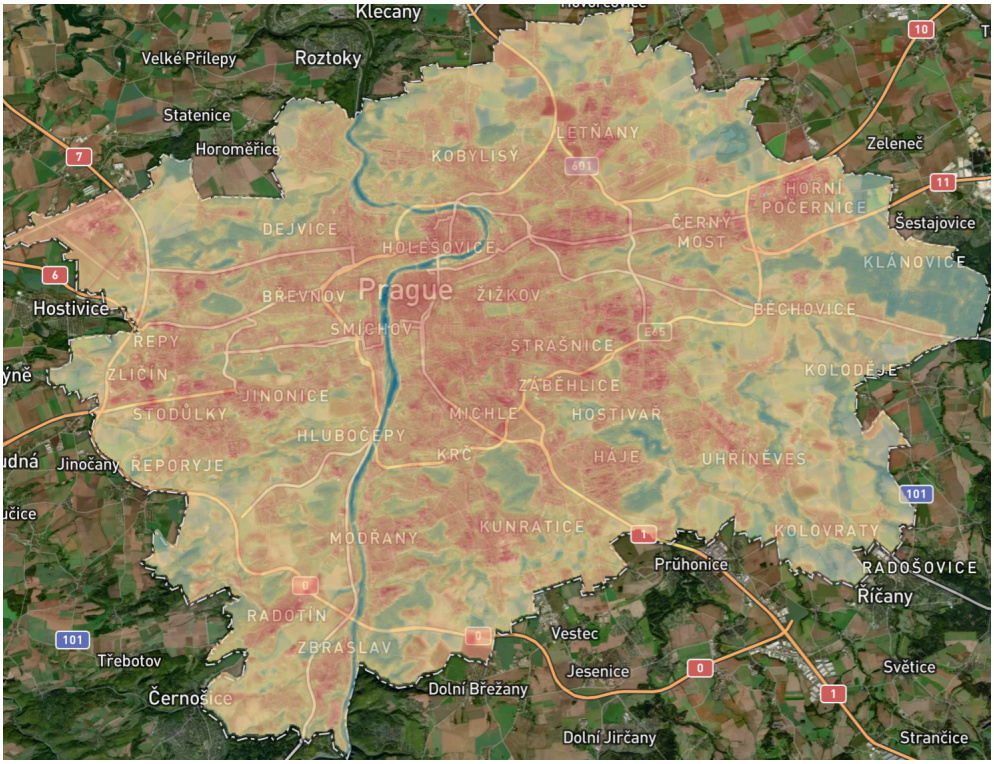

Figure 2.2 LST

background noise, offering more accurate vegetation assessments in densely vegetated or urban areas.2.4

Each of these indices utilizes specific spectral bands from satellite sensors to extract environmental information. The integration of multiple indices allows for a more nuanced understanding of urban ecosystems, enabling precise modeling of factors like vegetation health, surface heat distribution, and the presence of impervious surfaces. [14]

2.7 Remote Sensing and GIS Techniques for Data Preparation

Remote sensing and GIS techniques are critical for preparing satellite data for urban heat island (UHI) studies and machine learning (ML) analyses. Preprocessing steps, such as radiometric and geometric corrections, remove sensor errors and spatial distortions, ensuring data consistency across different time periods and sensors.[14] Cloud and shadow masking, often applied using algorithms like Fmask, further improve data quality, especially in optical imagery.[15]

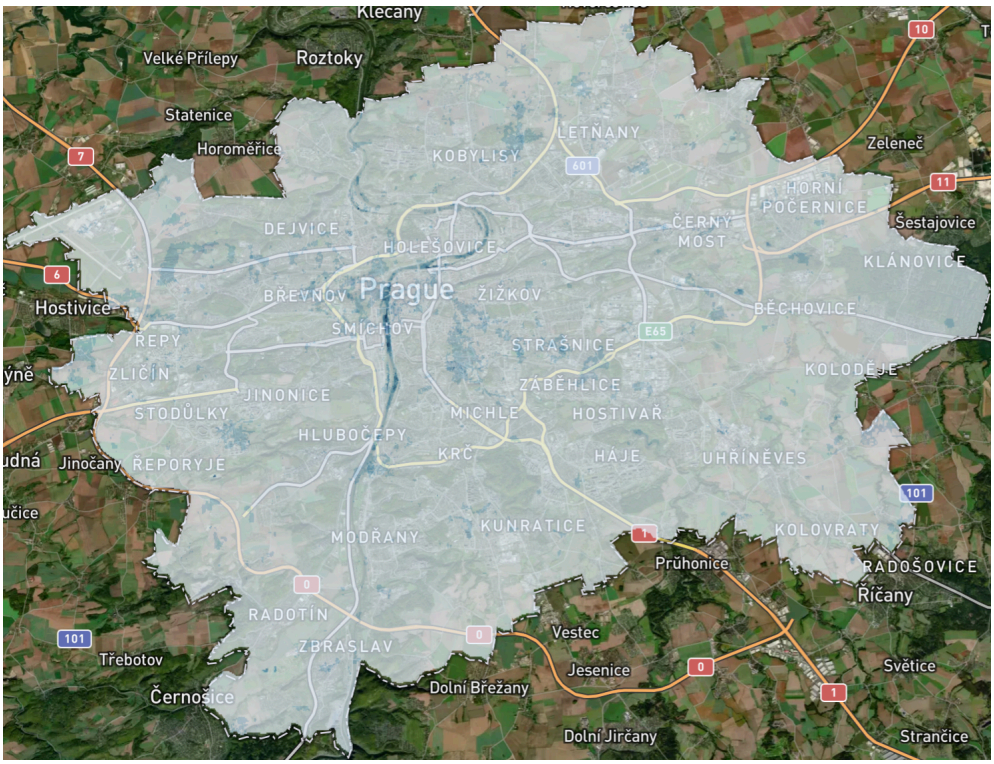

Figure 2.3 NDWI

Integrating multi-spectral and thermal data enables detailed environmental analysis. For example, multi-spectral bands are used to calculate indices like NDVI and NDBI, which help assess vegetation health and urban development[12], while thermal infrared bands provide land surface temperature (LST) data crucial for UHI mapping.[1]

GIS tools support data layering, feature extraction, and spatial analysis, converting raw satellite data into usable formats for ML models. Techniques like spatial resampling and zonal statistics help standardize datasets and extract meaningful patterns across urban environments. [11] These combined remote sensing and GIS methods ensure that data fed into predictive models are accurate, reliable, and actionable.

2.8 Urban Heat Islands: Formation and Characteristics

Urban Heat Islands (UHIs) are areas within cities that are significantly warmer than surrounding rural regions, primarily due to urbanization. Natural land covers are replaced with buildings, asphalt, and concrete—materials that absorb and retain heat more than vegetation or soil.

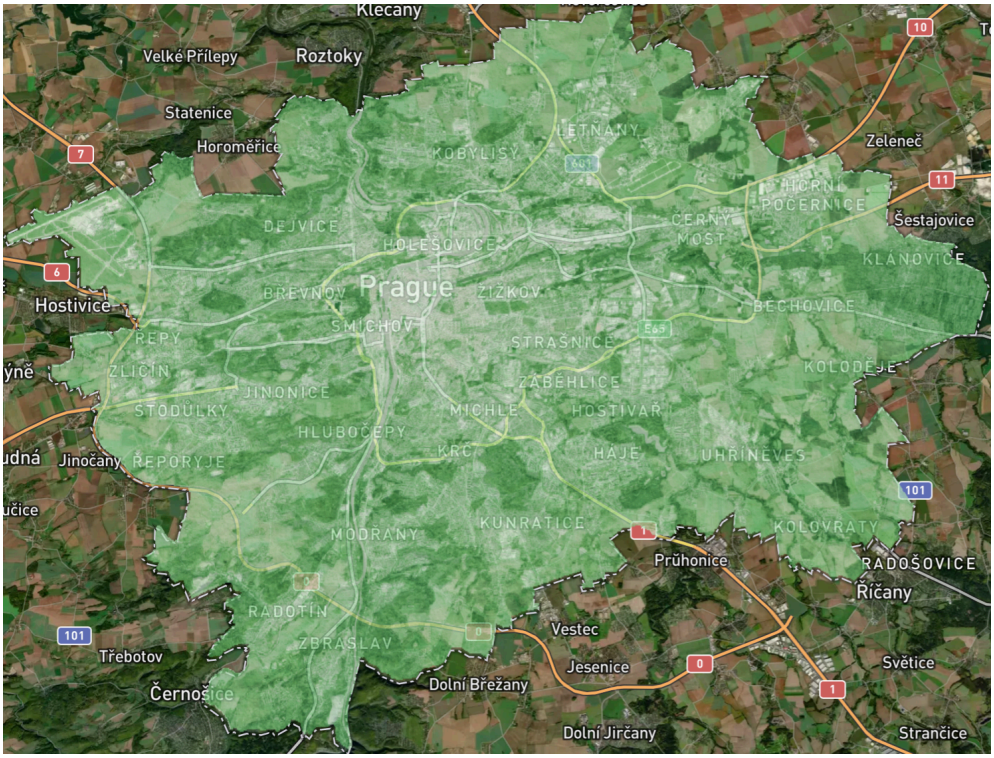

Figure 2.4 EVI

Several key factors contribute to UHI formation:

- **Low albedo surfaces:** Materials like asphalt store heat during the day and release it at night, keeping urban areas warmer.
- **Lack of vegetation:** Trees and plants cool the air through evapotranspiration. Their absence reduces natural cooling.
- **Anthropogenic heat:** Emissions from traffic, industry, and air conditioning increase local temperatures.
- **Urban geometry:** Narrow streets and tall buildings trap heat and block wind, creating "urban canyons."

The UHI effect is usually strongest at night and during summer. Within cities, temperature differences vary—green areas tend to be cooler, while dense built-up zones can become heat hotspots.

Chapter 3

Economic Impact Assessment Framework

Urban heat islands (UHIs) impose significant economic burdens on cities, affecting key sectors such as healthcare, energy consumption, and infrastructure maintenance. Machine learning (ML) models, combined with remote sensing and geospatial data, offer predictive capabilities that enable proactive decision-making to mitigate the economic impacts of extreme heat events. This section explores an economic impact assessment framework, integrating forecasting tools for urban heat mitigation, and evaluates how ML-based thermal data predictions can optimize infrastructure investments and resource allocation.

3.1 Key Economic Indicators Affected by Urban Heat Events

Urban heat events create cascading economic impacts across multiple sectors. This analysis categorizes the primary economic indicators that can be quantified to assess the benefits of predictive modeling

3.1.1 Energy Expenditures

Higher urban temperatures increase electricity demand, particularly for air conditioning, placing a financial strain on businesses and households.

Studies on urban redevelopment using ML-based thermal data suggest that energy-efficient infrastructure can reduce peak cooling demands by 15-20 percent. [15] [16]

Forecasting tools provide early warnings on heat intensity, allowing energy providers to adjust power grid loads and prevent blackouts, thus avoiding economic disruptions.

3.1.2 Infrastructure Maintenance and Longevity

Heat stress accelerates material degradation in roads, bridges, and buildings, requiring frequent repairs and increasing maintenance costs.

Remote sensing analysis of UHI in Quezon City, Philippines, shows that temperature variations reduce the lifespan of road materials by up to 30 percent due to thermal expansion and contraction cycles. [12]

If the asphalt grade is incorrect by one 6°C increment, the net present value of maintenance and rehabilitation costs can increase by approximately 6.8 to 9.8 percent depending on the road type. For two increments, these costs escalate to 9.2 to 17.4 percent. [17]

ML-based forecasting of high-risk zones enables targeted investment in heat-resistant materials and smart cooling infrastructure, extending urban infrastructure lifespan.

3.1.3 Healthcare Costs

Extreme heat increases heat-related illnesses such as heat strokes, cardiovascular stress, and respiratory complications, leading to higher hospital admissions and emergency response costs.

A study coming from Arizona State University from year 2016 estimates that the economic burden of heat-related illnesses in Phoenix is USD 479 million annually, representing 0.3 percent of the city's GDP. [1]

Predictive models integrating satellite-derived LST and socioeconomic data can help allocate healthcare resources efficiently, reducing emergency response delays and costs.

3.2 Energy savings

In the section below I provide an estimation of energy savings on air conditioners in the city of Prague using prediction model for city planning.

3.2.1 Estimation for Prague

Average energy consumption in Czechia was 25 million tonnes of oil equivalent (Mtoe). [18] Households accounted for approximately 15 percent of the total energy consumption, equating to 3.75 Mtoe. In European households, air conditioning typically represents about 1-5 percent of residential energy consumption. Assuming a 3 percent average, this amounts to 0.1125 Mtoe and since the air conditioners work during four summer months only it results in 0.0375 Mtoe for air conditioning in Czechia. As Prague comprises roughly 12 percent of Czechia's population, its share of air conditioning energy use would be approximately 0.0045 Mtoe, equivalent to 52 GWh.

Studies on urban redevelopment using ML-based thermal data suggest that energy-efficient infrastructure can reduce peak cooling demands by 15-20 percent. [15] [16]. Assuming 15 percent reduction to Prague's 52 GWh annual air conditioning energy use results in savings of approximately 7.8 GWh annually. Assuming an average electricity cost of €0.20 per kWh, the financial savings would be: 7,800,000 kWh * €0.20/kWh = €1,560,000 annually.

The whole is not getting rebuild in one day, but different areas of Prague are in active development so let's see how can we benefit by using ML in urban planning there.

3.2.2 Case scenario

In this section, I explore a case scenario focused on the Karlín area, which has been undergoing active redevelopment over the past decade, with projects still in progress. Figure 3.1 provides a heat map illustrating the area's urban heat dynamics.

Figure 3.1 Heat Map

3.2.2.1 Scenario A: without using ML model

Current situation of the development is that infrastructure changes lead to raising temperature of the area. That results in higher energy expenditures in the area during hot months and maintenance costs.

Estimated expenses only on air cooling systems in the are of Karlín are based on the calculations above around a million euros every year. Infrastructure maintenance costs aren't publicly available, we can estimate them at another million euros annually.

3.2.2.2 Scenario B: using ML model

Based on results of other studies on the topic both types of expenses can be reduced by 15 percent if taking actions in the development projects based on prediction models. That would result in €300,000 money spared every year just on air cooling and road maintenance in the area.

3.3 Environmental, Social, and Governance (ESG) Considerations

In addition to economic metrics, urban development strategies increasingly incorporate ESG — Environmental, Social, and Governance criteria to assess broader sustainability impacts. ESG frameworks are now a central component of both public policy and private investment, particularly in the context of climate change adaptation. Integrating ESG into the economic impact assessment of urban heat island (UHI) forecasting tools enhances their relevance, accountability, and alignment with the goals of the EU Corporate Sustainability Reporting Directive (CSRD) and global climate policy.

3.3.1 Environmental Perspective

The environmental benefits of using predictive models for UHI mitigation are multifold. By reducing heat stress through informed urban planning—such as increasing green cover, optimizing building materials, or improving cooling efficiency—cities directly contribute to:

- **Lower greenhouse gas emissions:** Predictive modeling enables energy-efficient interventions that decrease reliance on air conditioning, leading to reductions in electricity demand and associated CO emissions.
- **Improved urban biodiversity:** Strategic planting of vegetation and design of green corridors support urban ecosystems and biodiversity, mitigating the ecological fragmentation caused by dense infrastructure.
- **Sustainable land use:** Heat prediction helps in identifying climate-resilient zones, promoting development patterns that respect environmental thresholds and minimize ecological degradation.

These interventions align with the environmental component of ESG, reinforcing commitments to climate adaptation, resource efficiency, and environmental stewardship.

3.3.2 Social Perspective

From a social perspective, UHI disproportionately affects vulnerable populations, including the elderly, low-income households, and individuals with pre-existing health conditions. Incorporating social considerations into the predictive planning framework ensures that the benefits of climate resilience are equitably distributed:

- **Health equity:** Forecasting extreme heat enables targeted deployment of cooling centers, healthcare resources, and early warning systems to high-risk communities, reducing heat-related morbidity and mortality.
- **Affordable living conditions:** Lower energy expenditures through passive cooling and green infrastructure support cost savings for households, particularly those at risk of energy poverty.
- **Urban inclusivity:** Participatory planning based on spatial heat risk data fosters inclusive urban design, where citizen engagement informs interventions that reflect local needs.

3.3.3 Governance Perspective

The deployment of ML-based urban heat prediction tools requires transparent and accountable governance frameworks that address data ethics, implementation oversight, and long-term policy integration. ESG-aligned governance involves:

- **Transparent data use:** Ensuring that satellite-derived data and ML predictions are handled in compliance with privacy standards and open-data principles.
- **Public accountability:** Involving local authorities, city councils, and urban development agencies in the validation and oversight of model-driven planning decisions.
- **Long-term strategy alignment:** Integrating predictive models into broader climate action plans and sustainable development goals (SDGs), supported by institutional continuity and policy coherence.

Chapter 4

Business Case Development

The adoption of machine learning (ML)-based forecasting tools for predicting urban heat hotspots has significant implications for various stakeholders, including city planners, public health agencies, and energy providers. By leveraging thermal satellite data, these tools can optimize decision-making processes, improve resource allocation, and enhance urban resilience. This chapter outlines the potential applications of ML-driven urban heat prediction and estimates the Return on Investment (ROI) by analyzing the economic benefits in terms of operational cost reduction, planning efficiency, and public health improvements.

4.1 Stakeholder Applications of ML-Based Urban Heat Prediction

Stakeholders across various sectors stand to gain significantly from the adoption of ML-based urban heat prediction models. For city planners and municipal authorities, these models offer crucial insights into heat vulnerability zones, helping guide urban design choices and disaster preparedness strategies. By identifying at-risk areas, authorities can prioritize interventions such as increasing green spaces, improving building materials, and enhancing cooling infrastructure. Public health agencies also benefit by refining heatwave mitigation plans, ensuring that resources like cooling centers are effectively distributed and staffing is adjusted in anticipation of heat-related illnesses. Finally, energy providers leverage these models for better load forecasting, demand management, and infrastructure optimization, which ultimately leads to cost savings and improved service delivery. The integration of ML-driven tools across these stakeholders fosters more efficient, resilient, and equitable urban environments, driving both immediate and long-term benefits.

4.1.1 City Planners and Municipal Authorities

City planners can utilize ML-based heat forecasting models to optimize urban design and infrastructure planning. By identifying high-risk heat zones, authorities can implement targeted interventions such as increasing green spaces, using reflective roofing materials, and adjusting building regulations to promote heat resilience. Additionally, predictive models support disaster preparedness strategies by providing early warnings of extreme heat events, allowing municipalities to allocate emergency resources more efficiently. Integrating these forecasting tools into smart city initiatives enhances data-driven decision-making, leading to long-term improvements in urban sustainability. [19]

4.1.2 Public Health Agencies

Public health agencies benefit from ML-driven heat forecasting by improving heatwave mitigation plans. Predictive analytics assist in identifying high-risk zones, optimizing the placement of cooling centers, and ensuring equitable distribution of resources. By correlating heat exposure data with hospital admission records, healthcare institutions can anticipate spikes in heat-related illnesses and adjust staffing and emergency response measures accordingly. Advanced air quality monitoring, combined with thermal data, allows agencies to develop policies aimed at reducing the adverse health effects of prolonged heat exposure, particularly among vulnerable populations such as the elderly and low-income communities.

4.1.3 Energy Providers and Utility Companies

Energy providers can utilize urban heat prediction models to enhance load forecasting and demand management. Anticipating peak energy consumption during heatwaves allows for better grid stability and reduces the likelihood of power outages. Predictive models also help utilities optimize cooling system efficiency and adjust pricing strategies based on forecasted demand. Furthermore, insights from ML-driven thermal data enable infrastructure improvements, such as the deployment of energy-efficient technologies and smart grids, leading to long-term cost reductions and enhanced service reliability.

4.2 SWOT Analysis

In this section, I conduct a SWOT analysis to assess the strengths, weaknesses, opportunities, and threats associated with the use of satellite data for environmental modeling. Table 4.1 summarizes these factors.

Strengths	Weaknesses
Advanced Technology Utilization	High Initial Investment
Data Availability	Data Privacy and Security
Real-time Monitoring and Response	Dependency on External Data Sources
Opportunities	**Threats**
Expansion to Other Urban Areas	Technological Obsolescence
Enhancement of Smart City Initiatives	Regulatory and Compliance Issues

Table 4.1 SWOT Analysis

4.2.1 Strengths

- Advanced Technology Utilization: Leveraging state-of-the-art machine learning (ML) models, including CNNs and LSTMs, provides a sophisticated approach to analyzing complex spatial and temporal data from satellite imagery.
- Data Availability: Access to high-resolution, multi-temporal satellite data from reputable sources like NASA's Landsat and ESA's Sentinel enhances the accuracy and reliability of urban heat predictions.
- Real-time Monitoring and Response: The ability to monitor urban environments in real time allows for quick responses to heat events, potentially reducing public health risks and infrastructure damage.

4.2.2 Weaknesses

- High Initial Investment: Setting up the necessary infrastructure for ML and satellite data analysis can be costly, requiring significant upfront investment in technology and expertise.
- Data Privacy and Security: Handling sensitive geographical and environmental data necessitates robust security measures to prevent unauthorized access and data breaches.
- Dependency on External Data Sources: Reliance on satellite data from external agencies like NASA and ESA could pose risks related to data availability and control.

4.2.3 Opportunities

- Expansion to Other Urban Areas: Once proven effective, the model can be scaled and adapted to other cities worldwide, broadening the scope of impact.

- Enhancement of Smart City Initiatives: Integrating ML predictions with smart city technologies can enhance urban living standards by improving climate resilience and sustainability.

4.2.4 Threats

- Technological Obsolescence: Rapid advancements in technology could render the current ML models and methodologies outdated, necessitating continual updates and upgrades.
- Regulatory and Compliance Issues: Changes in data use policies and regulations could impact the availability and use of satellite data.

4.3 Risk assessment

In this section, we assess the potential risks associated with the use of satellite data and model development. Table 4.2 outlines key risks such as data inaccuracy, technological failure, compliance issues, and cybersecurity threats, along with their associated consequences, probabilities, and risk groups. The risk group matrix in Table 4.3 provides a clear categorization of the likelihood and severity of these risks, helping to prioritize mitigation efforts. To address these concerns, Table 4.4 presents proposed actions for risk removal and impact mitigation, which include enhanced validation protocols, system maintenance, compliance training, and the implementation of security measures. These tables provide a comprehensive overview of the risks and mitigation strategies, ensuring the reliability and security of the satellite data and models used in this research.

4.3.1 Risks

Risk Description	Consequences	Probability	Risk Group
Data Inaccuracy	4	2	3
Technological Failure	3	3	3
Regulatory Compliance Issues	3	2	2
Cybersecurity Threats	5	2	3
Data Privacy Breaches	4	2	3
Model Scalability Issues	3	2	2

Table 4.2 Risk Assessment

- Data Inaccuracy: Inaccurate or outdated satellite data could lead to erroneous predictions, potentially affecting decision-making processes.

Probability | Consequences	Insignificant (1)	Minor (2)	Moderate (3)	Major (4)
Almost Certain	1	2	3	3
Likely	1	2	2	3
Possible	1	2	2	3
Unlikely	1	2	2	2
Rare	1	1	1	2

Table 4.3 Risk Group Matrix

Risk Description	Removal Actions	Impact Mitigation Actions
Data Inaccuracy	Enhanced validation protocols	Regular data audits, calibration
Technological Failure	System maintenance	Fail-safes, backup systems
Compliance Issues	Continuous legal monitoring	Compliance training, consultations
Cybersecurity Threats	Use of advanced security measures	Regular security audits, updates
Data Privacy Breaches	Encryption, access controls	Data breach response plan
Model Scalability Issues	Incremental testing and upgrades	Scalability assessments

Table 4.4 Risk removal and impact mitigation

- Technological Failure: Breakdowns or malfunctions in hardware or software could interrupt data processing and model operation.
- Compliance Issues: Non-compliance with data use and privacy laws could lead to legal penalties and loss of reputation.
- Cybersecurity Threats: Unauthorized access to the system could compromise sensitive data and disrupt service operations.
- Data Privacy Breaches: Exposure of confidential data can violate privacy agreements and lead to legal challenges.
- Model Scalability Issues: Difficulty in scaling the model to larger or different urban environments could limit its applicability.

4.4 Expenses

In this section, a breakdown of the expenses associated with the project is provided. The costs cover a range of categories including data acquisition, computing resources, software licenses, personnel salaries, research and development, and operational overhead.

- Data Acquisition Costs: €20,000
- Computing Resources: €30,000
- Software Licenses: €15,000

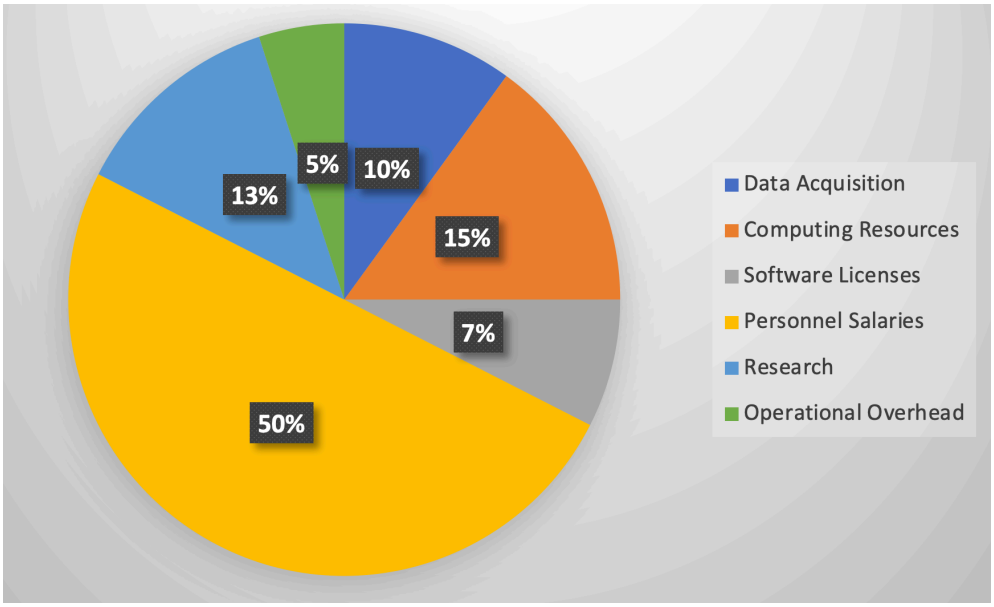

Figure 4.1 Expenses

- Personnel Salaries: €100,000
- Research and Development: €25,000
- Operational Overhead: €10,000

Figure 4.1 visualizes these expenses, offering a clear representation of how the budget is allocated across different areas.

- Data Acquisition Costs: Expenses related to obtaining satellite data from sources like NASA's Landsat and ESA's Sentinel.
- Computing Resources: Costs for computational power, including servers and cloud services necessary for processing large datasets and running ML models.
- Software Licenses: Fees for specialized software required for data processing, ML model development, and other tasks.
- Personnel Salaries: Salaries for data scientists, developers, project managers, and other staff involved in the project.
- Research and Development: Funds allocated to exploratory research, model testing, and validation.
- Operational Overhead: General administrative costs, including office space, utilities, and other miscellaneous expenses.

Annual Operating Costs:

Expense Category	Annual Cost (€)
Computing Resources	10,000
Personnel Salaries	30,000
Research and Development	5,000
Operational Overhead	5,000
Total	**50,000**

Table 4.5 Annual Operating Cost Breakdown

4.5 Financial projections

This section presents the financial projections for the ML-based urban heatmaps forecasting project over the next five years. The projections include detailed estimates of costs, revenue, and profits, illustrating the project's financial viability and potential return on investment (ROI). The initial investment for the project is substantial, totaling €200,000, primarily due to the costs associated with setting up the necessary infrastructure, acquiring data, and hiring skilled personnel.

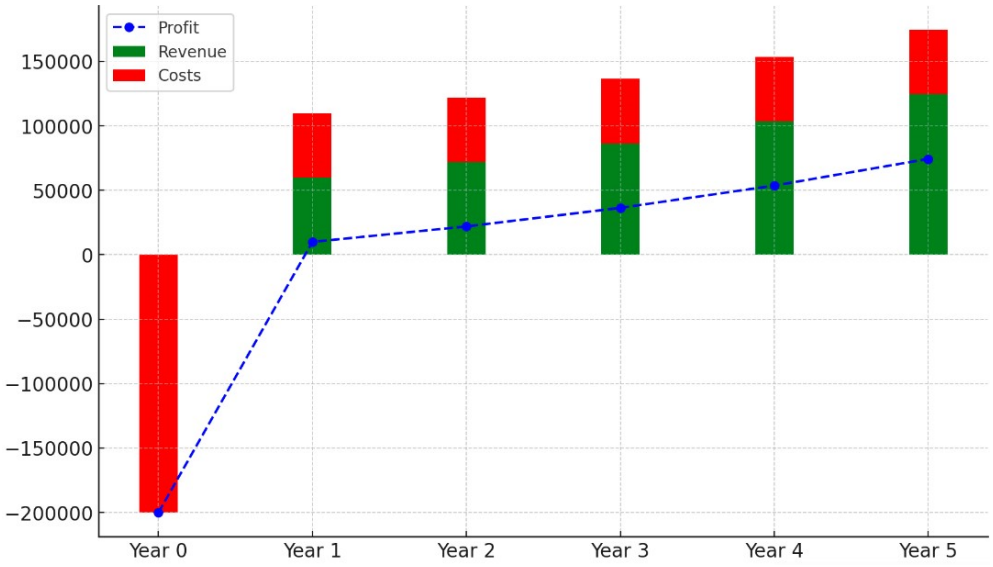

Figure 4.2 Financials

Revenue is expected to grow annually by 20 percent, as the service gains traction among municipalities and urban planners seeking to mitigate the effects of urban heat islands.4.2

Initial Investment: €200,000 (Year 0)4.1 Annual Operating Costs: €50,000 for subsequent years4.5 Annual Revenue: Starts at €60,000 in Year 1 and grows by 20 percent annually.4.6

Year	Costs (€)	Revenue (€)	Profit (€)
Year 0	-200,000	0	-200,000
Year 1	50,000	60,000	10,000
Year 2	50,000	72,000	22,000
Year 3	50,000	86,400	36,400
Year 4	50,000	103,680	53,680
Year 5	50,000	124,416	74,416

Table 4.6 Financial Projections

Chapter 5

Data Collection and Models Design

The primary source of training data for this research is satellite imagery from the Landsat and Sentinel missions. These datasets are critical for monitoring and analyzing various environmental factors, including urban heat islands, vegetation cover, and moisture content.

Figure 5.1 provides a chart illustrating the number of publications utilizing different satellite data sources in recent years, with the main contributors being Landsat and Sentinel. The chart highlights the growing trend of publications from these sources, which reflects the increasing reliance on satellite imagery in environmental research.

5.1 Bulk download

The acquisition of satellite imagery is conducted through an automated bulk download process, utilizing Python scripts to facilitate large-scale data retrieval. This process is supported by the EarthAccess API, which allows programmatic access to satellite data repositories maintained by NASA. The imagery is specifically sourced from the Harmonized Landsat Sentinel (HLS) dataset, including both HLS S30 (Sentinel-2) and HLS L30 (Landsat-8) products. For this study, only images with cloud coverage less than 30 percent are selected to ensure the quality and usability of the data for accurate environmental analysis.

The HLS S30 dataset from Sentinel-2 provides multi-spectral images that are crucial for extracting Enhanced Vegetation Index (EVI) and Normalized Difference Water Index (NDWI). These are derived using bands B02 (Blue), B03 (Green), B04 (Red), and B08 (Near-Infrared). Cloud masking is performed using the Fmask algorithm, which is integrated into the dataset processing pipeline to exclude cloud-contaminated pixels effectively.

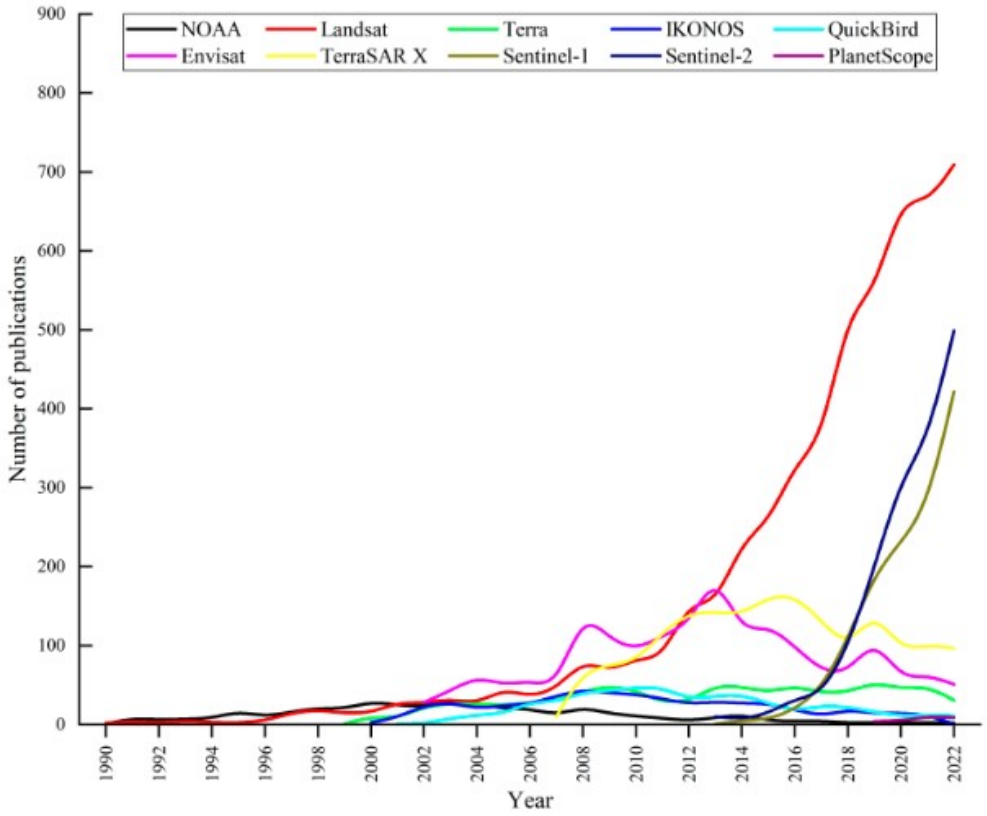

Figure 5.1 Sources (adapted from [14])

Similarly, the HLS L30 dataset from Landsat-8 includes thermal infrared data, particularly band B10, which is essential for calculating Land Surface Temperature (LST). This band, along with the Fmask cloud masking data, ensures that the temperature readings are precise and reliable.

5.2 Preprocessing

Once the raw satellite imagery in TIF format is collected, a series of preprocessing steps are performed to prepare the data for analysis. The first step involves cropping the images to the regions of interest to focus on the relevant urban areas. Following this, the individual satellite images are merged into composite datasets to ensure complete coverage and temporal consistency. The merging process is particularly important when using data from different satellite sources, as it helps address potential variations in sensor calibration and alignment between the images.

Data cleaning is the next crucial step, where the raw images are examined for any noise, inconsistencies, or artifacts that could affect the quality of the analysis. This involves removing clouds, correcting for atmospheric effects,

and filling in missing or corrupted data. The goal of the cleaning process is to ensure that the data is as accurate and reliable as possible before any further analysis is performed.

5.3 Indicators extraction

After preprocessing, various spectral indices are extracted from the cleaned satellite imagery. These indices serve as key indicators for environmental features, such as surface temperature, vegetation health, and moisture content. Specifically, I calculate thermal, vegetation, and moisture indices, which are integral for analyzing urban heat island effects, vegetation cover changes, and moisture distribution. These indices are then saved as PNG images with a defined resolution, providing a standardized format for subsequent analysis and model training.

The careful preparation of the satellite imagery ensures that the data is ready for feature extraction and machine learning analysis, enabling the development of predictive models that can accurately assess urban environmental conditions.

5.4 Model structure

The prototype model was developed using the Keras library within the TensorFlow framework. A range of model configurations was explored, varying in structural complexity, hyperparameters (e.g., learning rate, batch size), optimization techniques, and training duration. The iterative development process aimed to identify the architecture best suited for predicting high-quality LST maps from input environmental indicators.5.2

5.4.1 Model 1

The first of final models architecture incorporates the following key components:

Input Layers The model features three distinct input layers, each corresponding to grayscale images of LST, Enhanced Vegetation Index (EVI), and Normalized Difference Water Index (NDWI) indicators.

Shared Convolutional Architecture These inputs are processed through identical convolutional branches, enabling feature extraction for each indicator independently.

Output Layer The extracted features are combined to produce a single output: a 2D grayscale image representing the predicted LST map for the given input area.

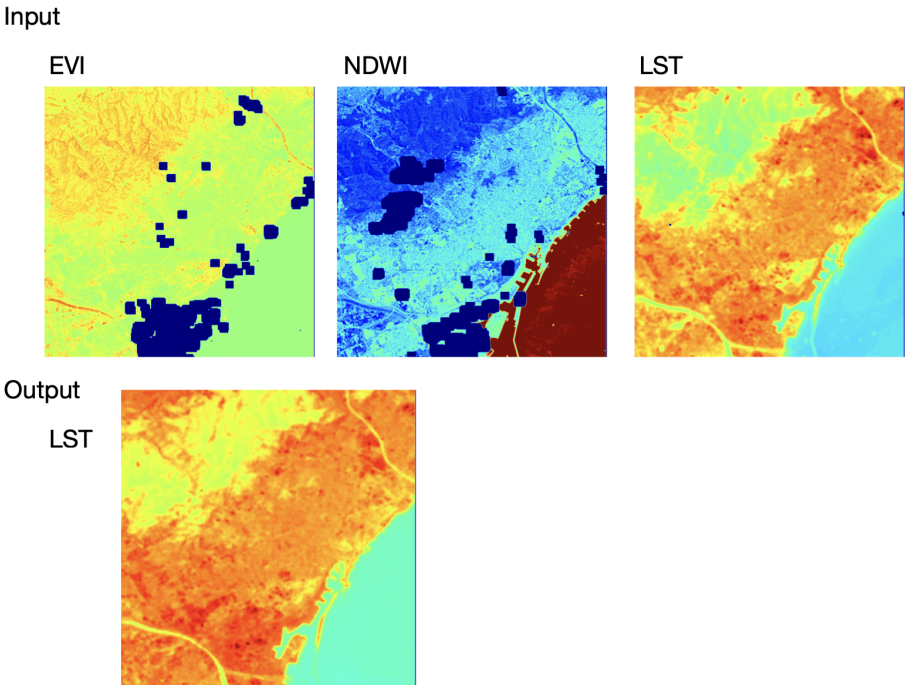

Figure 5.2 Input and Output

The model leverages convolutional neural networks (CNNs), which are well-suited for tasks involving spatial data such as images. CNNs demonstrated excellent performance in this application due to their ability to learn hierarchical spatial features. 5.3

5.4.2 Model 2

The second model was designed to address the complexities and resource-intensive demands of the initial prototype. Developed to optimize training speed and computational efficiency, this model employs a U-Net architecture, which is particularly renowned for its effectiveness in image segmentation tasks, a crucial aspect of our project's focus on predicting land surface temperature (LST) maps.

Input Layer The model starts with an input layer that accepts images of predefined shape, catering to the diverse dimensions of satellite imagery data.

Contracting Path The initial layers consist of a sequence of convolutions and max pooling operations:

- A convolutional layer with 16 filters of size 3x3, followed by a ReLU activation and same padding, ensures feature extraction without losing

input size.

- This is followed by a max pooling layer that reduces the dimensionality by half, enhancing the model's ability to handle larger images efficiently.

Bottleneck This central part of the network includes:

- A convolutional layer with 64 filters, maintaining the feature extraction momentum before the expansive path begins.

Expansive Path Sequentially increasing the spatial dimensions through up-sampling, each step is combined with a previous corresponding convolutional layer output:

- Up-sampling layers increase the resolution of the output feature maps, which are then concatenated with the corresponding feature maps from the contracting path, providing high-resolution features for precise localization.
- Convolutions following each up-sampling merge features to refine the predictions, culminating in finer details and improved accuracy.

Output Layer The final layer of the model uses a convolution to produce a single-channel output image, representing the predicted LST map with a sigmoid activation to ensure output values between 0 and 1, suitable for probability-based segmentation tasks.

This U-Net architecture significantly reduces training time and resource consumption without compromising the quality of the output. Its layered structure allows for efficient feature representation at multiple scales, which is critical for the accurate prediction of thermal patterns across urban landscapes. Below is the implementation code5.1 for this model, illustrating the layered approach used to construct the U-Net architecture. 5.4

```
def build_model(input_shape):
    inputs = Input(shape=input_shape)
    c1 = layers.Conv2D(16, (3, 3), activation='relu',
   padding='same')(inputs)
    p1 = layers.MaxPooling2D((2, 2))(c1)
    c2 = layers.Conv2D(32, (3, 3), activation='relu',
   padding='same')(p1)
    p2 = layers.MaxPooling2D((2, 2))(c2)
    c3 = layers.Conv2D(64, (3, 3), activation='relu',
   padding='same')(p2)
    u4 = layers.UpSampling2D((2, 2))(c3)
    u4 = layers.concatenate([u4, c2])
    c4 = layers.Conv2D(32, (3, 3), activation='relu',
   padding='same')(u4)
    u5 = layers.UpSampling2D((2, 2))(c4)
    u5 = layers.concatenate([u5, c1])
```

```
    c5 = layers.Conv2D(16, (3, 3), activation='relu',
padding='same')(u5)
    outputs = layers.Conv2D(1, (1, 1), activation='sigmoid')
(c5)
    model = models.Model(inputs, outputs)
    return model
```

Code listing 5.1 U-Net model

This streamlined model enhances our ability to quickly adjust and iterate on the forecasting process, allowing for rapid deployment and testing in various urban settings.

5.5 Training and Optimization

The model was compiled using the Adam optimizer and mean squared error (MSE) loss function, both of which are well-suited for regression tasks. Training was conducted over multiple epochs, with early stopping and model checkpointing to prevent overfitting.

```
train_ds, val_ds = load_data(DATA_DIR,
                batch_size=BATCH_SIZE, sequence_length=
SEQUENCE_LEN, sequence_step=SEQUENCE_STEP,
                future_step=FUTURE_STEP, target_size=
target_size)

model = build_model(input_shape=(IMAGE_X, IMAGE_Y,
INDICATORS_COUNT * SEQUENCE_LEN))
model.compile(optimizer='adam', loss='mse', metrics=['
mae'])

callbacks = [
    tf.keras.callbacks.ModelCheckpoint(
        filepath="models/checkpoints/model_checkpoint.h5
",
        save_best_only=True,
        monitor="val_loss",
        mode="min"
    ),
    tf.keras.callbacks.TensorBoard(log_dir="logs")
]

history = model.fit(
    train_ds,
    validation_data=val_ds,
    epochs=EPOCHS,
    callbacks=callbacks
)

```

```
25      model.save(MODEL_PATH)
```

Code listing 5.2 Training code

The training process utilized mixed-precision computation to accelerate performance on compatible hardware. 5.2

5.5.1 Averaged Output Labels

To improve the temporal consistency of the prediction target, the model was trained using averaged future thermal images instead of a single LST frame. This reduces the impact of short-term anomalies (e.g., cloud cover, noise) and provides a smoother supervisory signal during training.

The output label Y is computed as the mean of the next n LST frames following the input sequence:

$$Y = \frac{1}{n} \sum_{i=0}^{n-1} \mathrm{LST}_{t+\Delta+i}$$

Where t is the index of the last image in the input sequence, Δ is the future offset (prediction step), n is the number of future frames averaged.

5.6 Model Evaluation and Comparison Framework

To assess the performance of the trained deep learning models and to enable rigorous comparison across different configurations, an evaluation framework was implemented using a combination of statistical metrics and hotspot detection analysis. The evaluation plays a crucial role in selecting optimal model variants for deployment and understanding their robustness across different urban contexts.

5.6.1 Purpose of Evaluation

Given the large number of models trained—over 200 in total—each with unique configurations a consistent evaluation pipeline was necessary to benchmark and compare model accuracy, generalization, and suitability for operational deployment.

Configurations:

- Target city
- Resolution of images
- Sequence length and length of time steps between them
- Prediction horizon
- Batch size and number of training epochs

5.6.2 Evaluation Metrics

Each model was evaluated on a validation dataset using the following quantitative metrics:

- MAE (Mean Absolute Error): Measures the average pixel-wise difference between predicted and actual temperatures.
- MSE (Mean Squared Error): Quantify the magnitude of prediction errors.
- Cross-Correlation: Measures spatial alignment and similarity between images.
- Mutual Information (MI): Evaluates how much information is shared between prediction and ground truth images.
- F1 Score: Assesses the balance between precision and recall in identifying UHI hotspots.
- Recall: Measures the model's ability to correctly identify all true hotspot areas.

5.6.3 Hotspot Detection

A custom hotspot detection algorithm was developed to quantify the model's effectiveness in identifying extreme heat zones. It operates on a localized window and flags pixels that exceed a certain threshold over the neighborhood mean. This approach mimics real-world use cases where detecting anomalously hot zones is more valuable than merely reducing average error.

5.6.4 Evaluation Pipeline

An automated script was designed to train and evaluate models in batches using different configuration combinations. Results from all experiments were stored in a structured JSON format for further analysis. Each entry includes model parameters, training history, and average metric scores.

5.6.5 Model Selection

The results of the evaluation pipeline guided the selection of optimal model configurations for each city and use case. Models that demonstrated high hotspot recall and F1 scores, while maintaining low MAE and MSE, were prioritized for deployment in the web application and API interface.

The ability to benchmark models at scale across multiple spatial and temporal conditions proved critical in identifying trade-offs between model complexity, accuracy, and generalization. This evaluation system lays the foundation for future improvements, such as automated hyperparameter tuning or ensemble modeling.

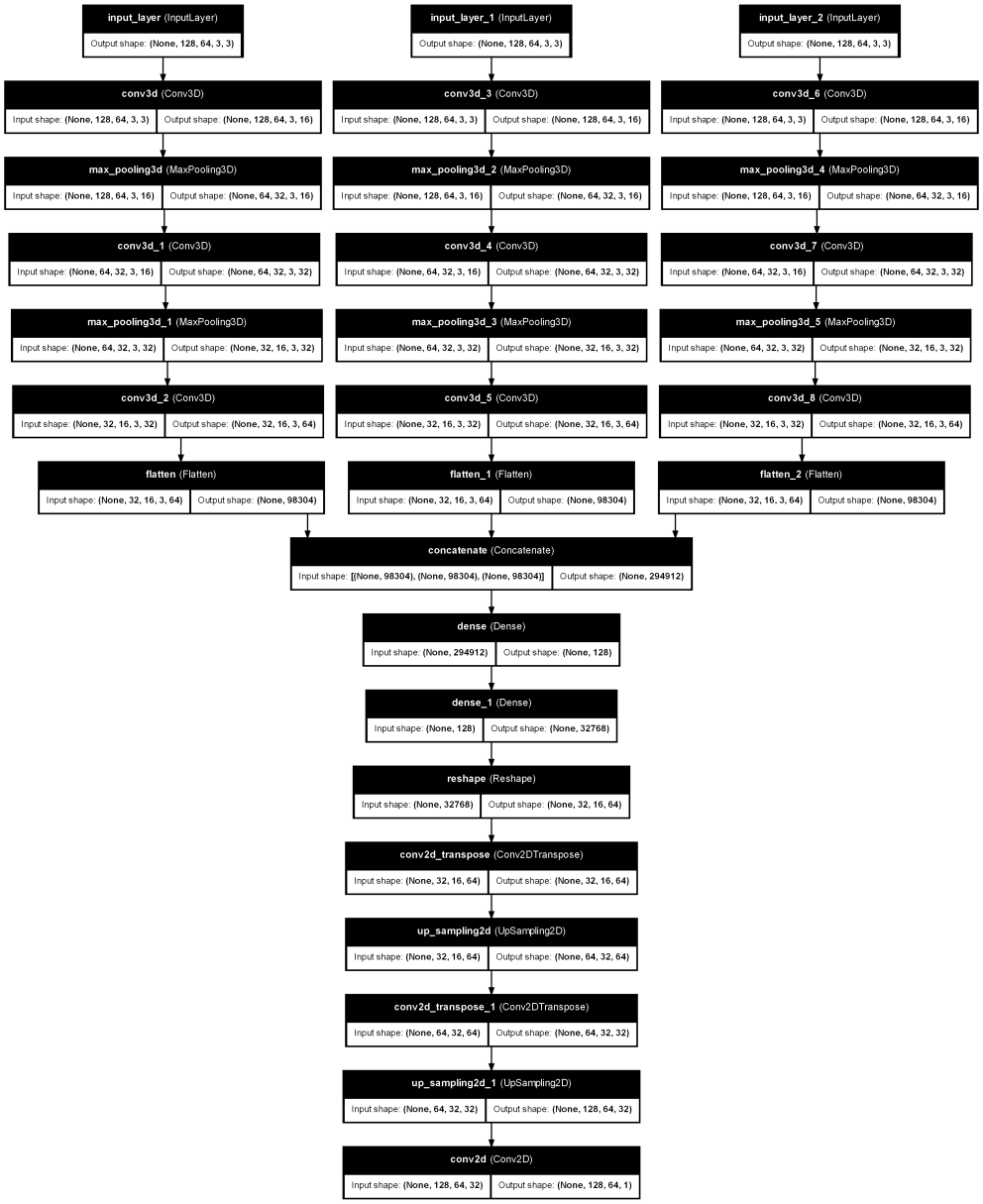

Figure 5.3 Model structure

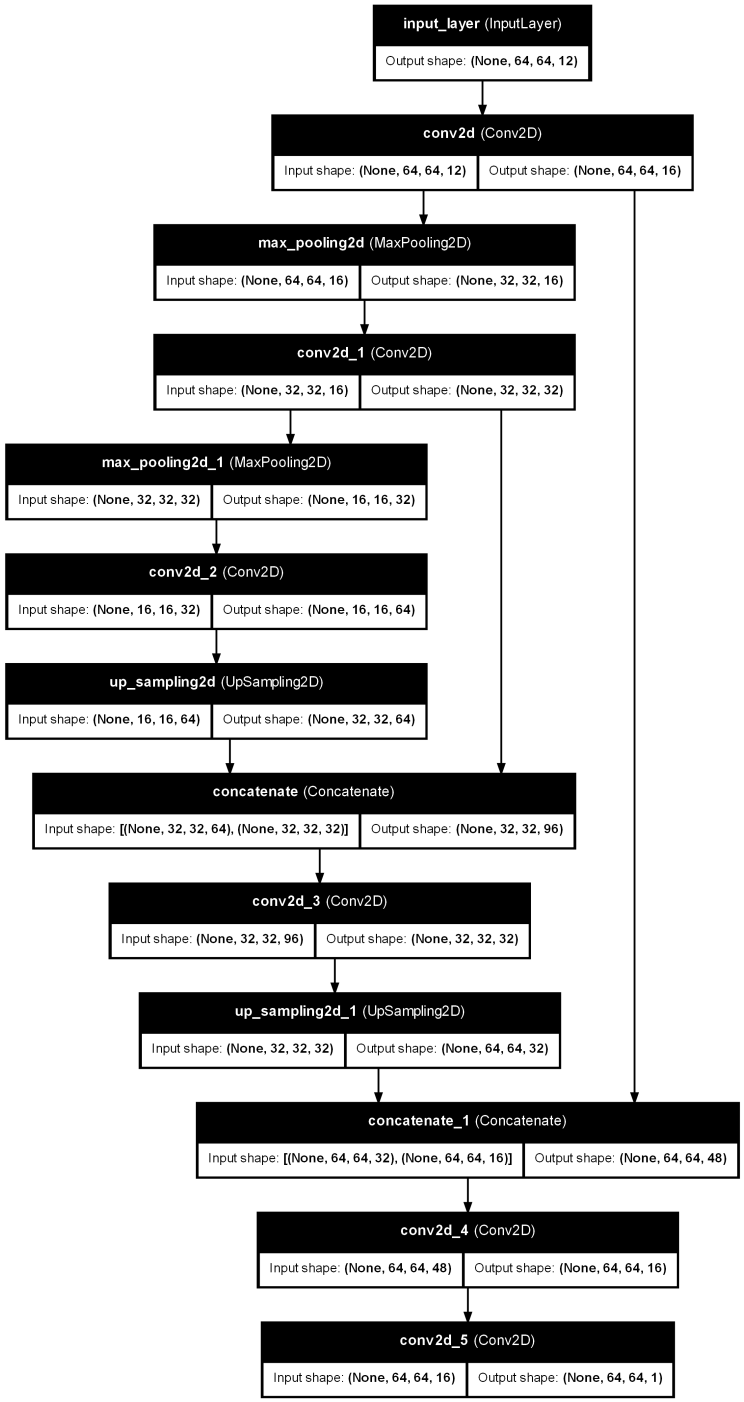

Figure 5.4 U-NET Model structure

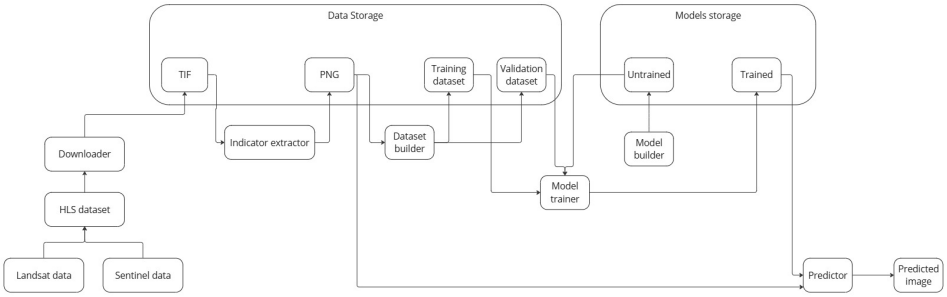

Figure 5.5 Structure

Chapter 6

Prototype Development

The development of this prototype involves selecting appropriate technologies and methodologies for implementing a robust deep learning model for predicting land surface temperature (LST) maps. The following sections detail the technologies employed and the model's development process. 5.5

6.1 Technologies used

In this section, I explore the various technologies employed in the creation of the machine learning prediction model for generating heat maps of urban areas. The decision to use these specific technologies was influenced by their functionality, availability of relevant features, and the familiarity I had with each. Ultimately, the aim was to select the most appropriate tools that would facilitate the creation of a high-performing, scalable model.

6.1.1 Python

Python was chosen as the primary programming language for this project due to its versatility and the vast ecosystem of libraries and frameworks it offers. Python is particularly well-suited for tasks involving machine learning, image processing, and tensor manipulation, which form the foundation of this work. Its intuitive syntax and active community support further reinforced its selection.

6.1.2 TensorFlow and Keras

TensorFlow served as the primary framework for developing and training the machine learning models. TensorFlow's extensive suite of features, including efficient tensor operations, optimization utilities, and hardware acceleration, made it an ideal choice for this project. Within the TensorFlow ecosystem,

Keras was utilized for building the model architecture. Keras offers a high-level API that simplifies the creation of complex neural network structures while providing flexibility and compatibility with TensorFlow's powerful backend.

6.1.3 NumPy and Pandas

NumPy was used for efficient numerical computations, while Pandas facilitated structured data handling and preprocessing, particularly during the preparation of metadata and timestamps associated with satellite imagery.

6.1.4 Matplotlib and Seaborn

These libraries were employed for visualizing both raw input data and model output. Visual inspections played a crucial role in validating the accuracy of predictions and identifying misalignments.

6.1.5 EarthAccess and Copernicus APIs

Automated scripts used NASA EarthAccess and Copernicus Open Access Hub APIs to fetch bulk satellite data, including thermal and vegetation indices for selected urban regions.

6.2 Model Access and End-User Interface

To ensure accessibility and usability of the predictive outputs, the system provides two complementary interface options for end-users: a programmatic Application Programming Interface (API) for automated access and integration, and an interactive web-based visualization portal for intuitive exploration of predictions. These interfaces are designed to support both technical users and decision-makers across urban planning, environmental monitoring, and climate resilience domains.

6.2.1 API for Programmatic Access

A RESTful API enables external users or systems to interact directly with the trained machine learning models. Through the API, users can retrieve existing predictions or generate new ones using their own input data. This method is particularly suitable for integration into smart city platforms, planning dashboards, and research workflows that require automation or large-scale processing.

The API exposes several endpoints, including:

- /predict — Allows users to submit custom environmental indicator data (e.g., NDWI, EVI, or thermal bands) and returns a predicted land surface temperature (LST) map as output. Input data must conform to a standardized format to ensure compatibility with the model pipeline.
- /results/<region>/<timestamp> — Provides access to precomputed prediction outputs for supported urban areas and time frames, enabling rapid retrieval of historical forecasts.
- /metadata — Returns descriptive metadata associated with each prediction, such as model version, resolution, input sources, and confidence metrics, supporting transparency and reproducibility.

This API facilitates real-time or batch-mode inference, allowing the system to be embedded into larger infrastructures for operational use or academic analysis.

6.2.2 Web Application for Interactive Visualization

For users seeking a more accessible and visually-oriented interface, a web application portal is available. The portal offers an interactive environment for viewing, comparing, and analyzing the predicted heat maps generated by the system. It is tailored to meet the needs of urban designers, climate policy advisors, and non-technical stakeholders.6.1

The key functionalities of the portal include:

- Geospatial Heat Map Viewer — Predicted LST maps are rendered as overlays on top of geographic basemaps, allowing users to visually identify spatial patterns of urban heat intensity across city districts.
- Temporal Navigation Tools — A timeline slider allows users to explore the evolution of heat patterns over time, supporting seasonal trend analysis and before/after comparisons for redevelopment scenarios.
- Custom Input Upload — Advanced users have the option to upload their own multi-band indicator images, which are processed by the model to generate customized LST predictions that are then displayed on the map.
- Downloadable Assets — Users can export prediction results in standard formats (e.g., PNG or GeoTIFF), accompanied by metadata files for documentation or further GIS analysis.

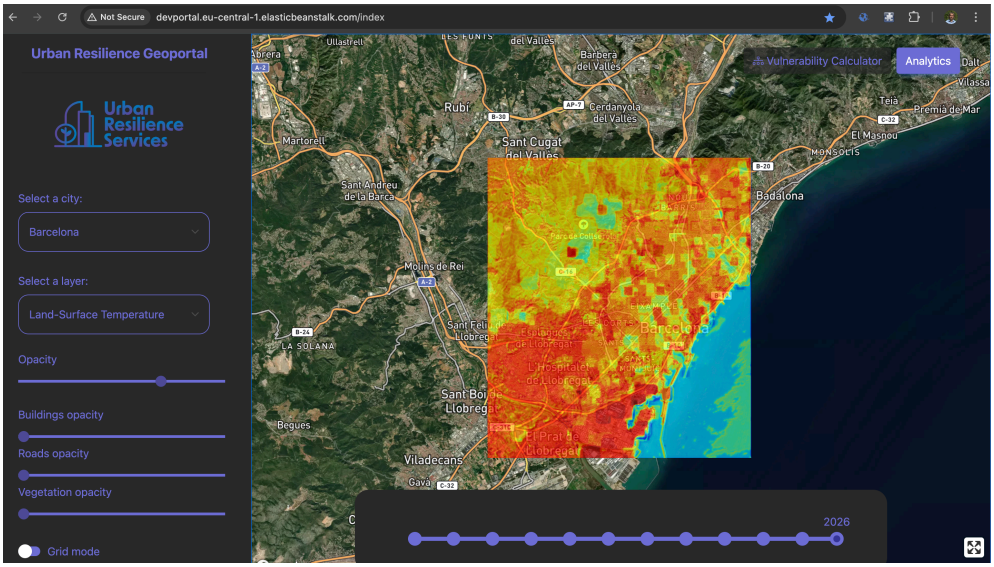

Figure 6.1 Geoportal

Chapter 7

Prototype Demonstration and Feedback

7.1 Presentation

7.1.1 Comparison of predictions with actual data

To assess the validity of the developed forecasting models, visual and numerical comparisons were made between the predicted land surface temperature (LST) maps and corresponding real-world satellite observations. In various test cases across different cities, the predictions closely mirrored the actual heat distributions, particularly in identifying spatial patterns of urban heat hotspots. The visual resemblance between prediction and reality, as presented in Figures 7.1 and 7.2, supports the reliability of the model's outputs.

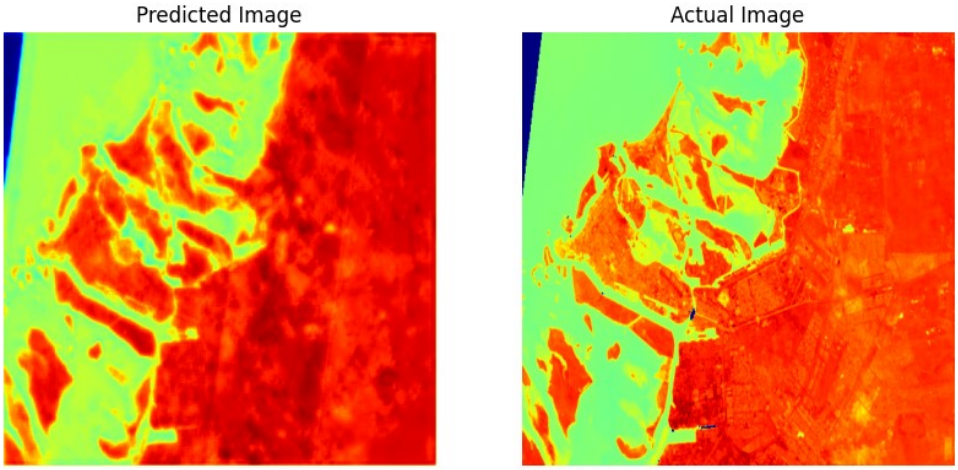

Figure 7.1 Abudhabi

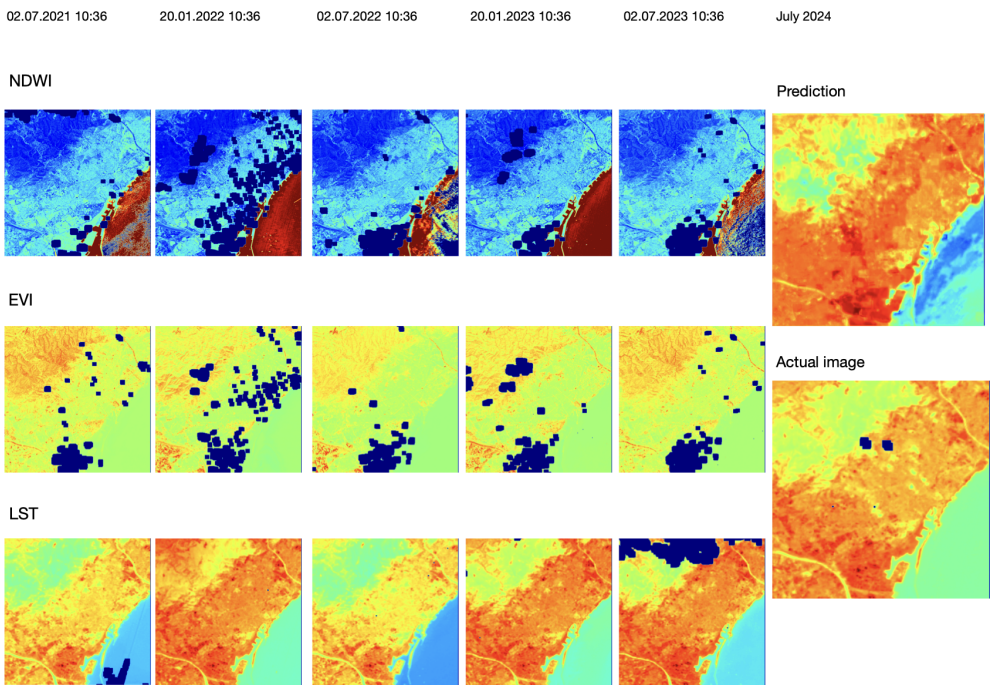

Figure 7.2 Barcelona prediction

7.1.2 Quantitative Evaluation of Prediction Accuracy

To evaluate the performance of the machine learning models developed in this project, a quantitative assessment was conducted using key image similarity and classification metrics. These metrics were applied to compare the predicted land surface temperature (LST) maps with actual thermal imagery for selected cities: Amsterdam, Barcelona, and Casablanca. The evaluation focused on both pixel-level similarity and hotspot detection accuracy, offering insights into model performance across different urban and climatic contexts. The table below presents the best-performing model metrics for each city.

Table 7.1 Evaluation metrics of best-performing model per city

Metric	Amsterdam	Barcelona	Casablanca
MAE	0.039	0.099	0.124
MSE	0.005	0.018	0.026
Cross-Correlation	0.368	0.648	0.598
Mutual Information	0.302	1.817	1.638
F1 Score	0.225	0.343	0.390
Recall	0.267	0.323	0.380

7.2 Feedback

7.2.1 Ing. David Buchtela, Ph.D.

Ing. David Buchtela, Ph.D., head of the Management Informatics specialization at our faculty, provided constructive and encouraging feedback during the thesis presentation. He expressed a strong appreciation for the comprehensive economic and business analysis, particularly its clarity in communicating the real-world applicability of the forecasting system to various stakeholders such as city planners, utility providers, and healthcare institutions.

In addition to affirming the relevance of the business case, Dr. Buchtela proposed an insightful extension of the model's potential use cases. He highlighted the applicability of urban heat forecasting in the real estate sector, emphasizing its utility for both individuals and real estate agencies during property acquisition decision-making. By integrating predicted long-term thermal development into property evaluation, users of the visual interface can identify areas likely to experience more favorable microclimates in the future, potentially reducing energy consumption and improving living comfort.

Dr. Buchtela's feedback underscores the model's versatility and relevance beyond public infrastructure and policy planning. It highlights the potential for wider adoption in the private sector and introduces a novel dimension of climate-informed decision-making in real estate economics—thereby broadening the scope of the tool's impact.

Chapter 8

Conclusion and Future Directions

This research embarked on an ambitious journey to leverage advanced machine learning (ML) techniques and extensive satellite data for predicting urban heat islands (UHIs), a pressing issue exacerbated by urban expansion and climate change. The use of convolutional neural networks (CNNs) and long short-term memory (LSTM) networks allowed for the extraction and analysis of complex spatial and temporal patterns from NASA's Landsat and ESA's Sentinel datasets, providing unprecedented insights into urban heat dynamics.

8.1 Conclusion

The developed ML models demonstrated significant potential in forecasting extreme heat events, offering a robust tool for urban planners and policymakers to proactively address the challenges posed by increased urban temperatures. By integrating geospatial analysis with machine learning, this research contributed to the advancement of urban climatology and demonstrated the utility of satellite data in environmental monitoring and planning.

The economic analysis outlined the potential savings and cost reductions in energy consumption, healthcare, and infrastructure maintenance by implementing ML-driven UHI mitigation strategies. These findings underscore the relevance and urgency of adopting advanced technologies to foster sustainable urban environments.

8.1.1 Model Evaluation Summary

Quantitative evaluation of the best-performing models revealed diverse strengths across different cities. Models for Amsterdam achieved the lowest mean absolute error (MAE), showcasing excellent pixel-level prediction accuracy. In

contrast, models developed for Barcelona and Casablanca exhibited superior performance in classification metrics, such as F1 Score and cross-correlation, indicating higher efficacy in detecting and aligning spatial patterns of urban heat hotspots.

Figure 8.1 presents a comparative view of key metrics across cities, highlighting trade-offs between precise value prediction (MAE) and hotspot detection accuracy (F1 Score, Cross-Correlation). These results suggest that optimal model performance depends on the intended application—whether minimizing thermal error or accurately identifying extreme heat zones.

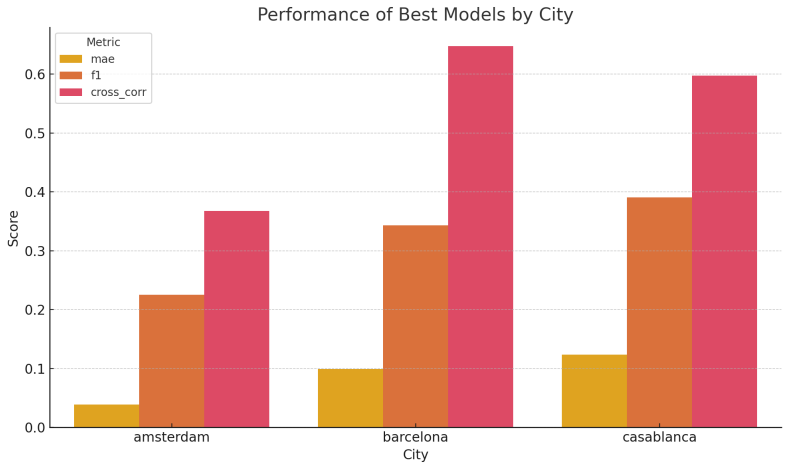

Figure 8.1 Comparison of models across cities

These findings underscore the importance of city-specific model tuning and the potential of deep learning to address urban thermal challenges with both precision and contextual awareness.

8.2 Contributions

This thesis has made several key contributions to the field of urban climatology and machine learning:

- Development of a predictive model: By employing neural networks, the project successfully created a reliable model capable of predicting urban heat maps with high accuracy.
- Economic impact analysis: The study quantitatively demonstrated the potential economic benefits of implementing ML-based solutions for urban heat management, highlighting significant opportunities for cost savings.
- Advancement in remote sensing applications: The use of satellite data for environmental monitoring was expanded, showcasing the effectiveness of remote sensing in urban planning.

8.3 Future Work

Looking ahead, several avenues for future research and development can be considered to enhance the effectiveness and applicability of the findings:

- Model refinement and optimization: Further tuning and testing of the ML models to improve accuracy and reduce computational demands. Exploration of newer architectures and hybrid models may provide breakthroughs in predictive performance.
- Expansion to other urban areas: Applying the developed models to a broader range of cities and climates to test their generalizability and refine them based on diverse urban layouts and environmental conditions.
- Integration with IoT devices: Developing a real-time monitoring system by integrating the predictive models with Internet of Things (IoT) devices across urban centers could provide live data feeds, enhancing model accuracy and timeliness.
- Public policy and community engagement: Working closely with urban planners and policymakers to implement findings and models in real-world scenarios. This includes community engagement to raise awareness about UHI effects and mitigation strategies.
- Socioeconomic impact studies: Conducting detailed studies on the socioeconomic impacts of UHI mitigation strategies could further justify investment in such technologies and encourage broader adoption.

The potential of machine learning in transforming urban environmental management has just begun to be tapped. As technology advances and more data becomes available, the opportunities to enhance urban living through intelligent data analysis and application will only grow, promising a cooler, more sustainable future for urban environments worldwide.

Appendix A
Source Code

This chapter contains the key Python source files used in the project, including model training, evaluation, and prediction.

A.1 Training Script – `train.py`

```
import os
import pickle
import pandas as pd
import matplotlib.pyplot as plt
import json

import tensorflow as tf
from tensorflow.keras.utils import plot_model
from tensorflow.keras.mixed_precision import
    set_global_policy

from utils.data_loader import load_data
from utils.models import build_model


if __name__ == "__main__":

    with open('config.json', 'r') as file:
        config = json.load(file)

    MODEL_PATH = config['MODEL_PATH']
    DATA_DIR = config['DATA_DIR']
    IMAGE_Y = config['IMAGE_Y']
    IMAGE_X = config['IMAGE_X']
    SEQUENCE_LEN = config['SEQUENCE_LEN']
    SEQUENCE_STEP = config['SEQUENCE_STEP']
    FUTURE_STEP = config['FUTURE_STEP']
```

```
27      INDICATORS_COUNT = config['INDICATORS_COUNT']
28      BATCH_SIZE = config['BATCH_SIZE']
29      EPOCHS = config['EPOCHS']
30
31      target_size=(IMAGE_X, IMAGE_Y)
32
33      train_ds, val_ds = load_data(DATA_DIR,
34                      batch_size=BATCH_SIZE, sequence_length=
    SEQUENCE_LEN, sequence_step=SEQUENCE_STEP,
35                      future_step=FUTURE_STEP, target_size=
    target_size)
36
37      model = build_model(input_shape=(IMAGE_X, IMAGE_Y,
    INDICATORS_COUNT * SEQUENCE_LEN))
38      plot_model(model, to_file="model_structure.png",
    show_shapes=True, show_layer_names=True)
39
40      model.compile(optimizer='adam', loss='mse', metrics=['
    mae'])
41
42      callbacks = [
43          tf.keras.callbacks.ModelCheckpoint(
44              filepath="models/checkpoints/model_checkpoint.h5
    ",
45              save_best_only=True,
46              monitor="val_loss",
47              mode="min"
48          ),
49          tf.keras.callbacks.TensorBoard(log_dir="logs")
50      ]
51
52      set_global_policy('mixed_float16')
53
54      history = model.fit(
55          train_ds,
56          validation_data=val_ds,
57          epochs=EPOCHS,
58          callbacks=callbacks
59      )
60
61      os.makedirs("models/final", exist_ok=True)
62      model.save(MODEL_PATH)
63
64      history_df = pd.DataFrame(history.history)
65      history_df.to_csv("training_log.csv", index=False)
66
67      with open("training_history.pkl", "wb") as f:
68          pickle.dump(history.history, f)
```

Code listing A.1 Model Training Script

A.2 Prediction Script – predict.py

```
import numpy as np
import os
import json

import tensorflow as tf
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing.image import
    array_to_img

from utils.data_loader import preprocess_image,
    apply_colormap


def prepare_input_sequences(data_dir, sequence_length=3,
    sequence_step=1, target_size=(128, 128)):
    def get_sorted_image_paths(indicator_dir):
        png_dir = os.path.join(indicator_dir, "png")
        year_dirs = sorted(os.listdir(png_dir))
        image_paths = []
        for year in year_dirs:
            year_path = os.path.join(png_dir, year)
            if os.path.isdir(year_path):
                year_images = sorted(
                    [os.path.join(year_path, img) for img in
     os.listdir(year_path) if img.endswith(('.png', '.jpg'))]
                )
                image_paths.extend(year_images)
        return image_paths

    evi_dir = os.path.join(data_dir, "evi")
    ndwi_dir = os.path.join(data_dir, "ndwi")
    lst_dir = os.path.join(data_dir, "lst")

    evi_paths = get_sorted_image_paths(evi_dir)
    ndwi_paths = get_sorted_image_paths(ndwi_dir)
    lst_paths = get_sorted_image_paths(lst_dir)

    min_length = min(len(evi_paths), len(ndwi_paths), len(
    lst_paths))
    total_samples = min_length - (sequence_length - 1) *
    sequence_step

    if total_samples < 1:
        raise ValueError(
            f"Not enough images to form sequences. Required
    at least {sequence_length * sequence_step}, found: {
    min_length}"
```

```
        )

    input_sequences = []

    for i in range(total_samples):
        evi_sequence = [preprocess_image(evi_paths[j],
    target_size) for j in range(i, i + sequence_length *
    sequence_step, sequence_step)]
        ndwi_sequence = [preprocess_image(ndwi_paths[j],
    target_size) for j in range(i, i + sequence_length *
    sequence_step, sequence_step)]
        lst_sequence = [preprocess_image(lst_paths[j],
    target_size) for j in range(i, i + sequence_length *
    sequence_step, sequence_step)]

        evi_stack = tf.concat(evi_sequence, axis=-1)
        ndwi_stack = tf.concat(ndwi_sequence, axis=-1)
        lst_stack = tf.concat(lst_sequence, axis=-1)

        input_tensor = tf.concat([evi_stack, ndwi_stack,
    lst_stack], axis=-1)
        input_sequences.append(input_tensor)

    return np.array(input_sequences)


def predict_and_save(model_path, data_dir, output_dir,
    sequence_length=3, sequence_step=1, target_size=(128,
    128)):
    model = load_model(model_path)
    input_sequences = prepare_input_sequences(data_dir,
    sequence_length, sequence_step, target_size)

    predictions = model.predict(input_sequences)

    os.makedirs(output_dir, exist_ok=True)
    for i, pred in enumerate(predictions):
        pred_img = array_to_img(pred)
        pred_img = apply_colormap(pred_img)
        pred_img = array_to_img(pred_img)
        pred_img.save(os.path.join(output_dir, f"
    predicted_image_{i + 1}.png"))

if __name__ == "__main__":
    OUTPUT_DIR = "predicted_images"

    with open('config.json', 'r') as file:
        config = json.load(file)


```

```
    MODEL_PATH = config['MODEL_PATH']
    DATA_DIR = config['DATA_DIR']
    IMAGE_Y = config['IMAGE_Y']
    IMAGE_X = config['IMAGE_X']
    SEQUENCE_LEN = config['SEQUENCE_LEN']
    SEQUENCE_STEP = config['SEQUENCE_STEP']
    target_size = (IMAGE_X, IMAGE_Y)

    predict_and_save(MODEL_PATH, DATA_DIR, OUTPUT_DIR,
    sequence_length=SEQUENCE_LEN, sequence_step=SEQUENCE_STEP
    , target_size=target_size)
```

Code listing A.2 Prediction Script

A.3 Evaluation Script – evaluation.py

```
import tensorflow as tf
import numpy as np
import os
import json
import re
from tensorflow.keras.models import load_model
from sklearn.metrics import mean_absolute_error, f1_score,
    recall_score, mean_squared_error
from skimage.metrics import peak_signal_noise_ratio as
    psnr_metric
from scipy.stats import pearsonr
from sklearn.metrics import mutual_info_score

from utils.data_loader import load_city_data



def detect_hotspots(image, window_size=10, threshold_factor
    =1.10):

    height, width = image.shape
    hotspot_mask = np.zeros_like(image, dtype=np.uint8)
    padded_image = np.pad(image, pad_width=((window_size//2,
     window_size//2), (window_size//2, window_size//2)),
                          mode='reflect')

    for i in range(height):
        for j in range(width):
            local_window = padded_image[i:i+window_size, j:j
    +window_size]

            local_window = local_window.flatten()
```

```
            local_window = np.delete(local_window, len(local_window)//2)

            local_avg = np.mean(local_window)

            if image[i, j] > local_avg * threshold_factor:
                hotspot_mask[i, j] = 1

    return hotspot_mask

def cross_correlation(a, b):
    a_mean = np.mean(a)
    b_mean = np.mean(b)
    numerator = np.sum((a - a_mean) * (b - b_mean))
    denominator = np.sqrt(np.sum((a - a_mean)**2) * np.sum((b - b_mean)**2))
    return numerator / denominator if denominator != 0 else 0


def mutual_information(a, b, bins=256):
    hist_2d, _, _ = np.histogram2d(a.flatten(), b.flatten(), bins=bins)
    return mutual_info_score(None, None, contingency=hist_2d)


def evaluate_model(model, data_dir, sequence_length=3, sequence_step=1, future_step=1,
                   target_size=(128, 128), hotspot_threshold_factor=1.05):
    train_data, _ = load_city_data(data_dir, batch_size=1, sequence_length=sequence_length,
                                   sequence_step=sequence_step, future_step=future_step, target_size=target_size)

    metrics = {
        "mae": [],
        "mse": [],
        "rmse": [],
        "psnr": [],
        "cross_corr": [],
        "mutual_info": [],
        "f1": [],
        "recall": [],
    }

    hotspot_images = 0
    total_images = 0

```

```
    image_index = 0

    for inputs, actual_output in train_data:

        predicted_output = model.predict(inputs, verbose=0)

        pred_gray = predicted_output[0, :, :, 0].numpy() if isinstance(predicted_output, tf.Tensor) else predicted_output[0, :, :, 0]
        actual_gray = actual_output[0, :, :, 0].numpy() if isinstance(actual_output, tf.Tensor) else actual_output[0, :, :, 0]

        # --- Basic Metrics ---
        mae = mean_absolute_error(actual_gray.flatten(), pred_gray.flatten())
        mse = mean_squared_error(actual_gray.flatten(), pred_gray.flatten())
        rmse = np.sqrt(mse)
        psnr = psnr_metric(actual_gray, pred_gray, data_range=1.0)
        xcorr = cross_correlation(actual_gray, pred_gray)
        mi = mutual_information(actual_gray, pred_gray)

        # --- Hotspot Detection ---
        actual_hotspots = detect_hotspots(actual_gray, threshold_factor=hotspot_threshold_factor)
        pred_hotspots = detect_hotspots(pred_gray, threshold_factor=hotspot_threshold_factor)

        f1 = f1_score(actual_hotspots.flatten(), pred_hotspots.flatten(), zero_division=0)
        recall = recall_score(actual_hotspots.flatten(), pred_hotspots.flatten(), zero_division=0)

        if np.sum(actual_hotspots) > 0:
            hotspot_images += 1
        total_images += 1

        # --- Save Metrics ---
        metrics["mae"].append(mae)
        metrics["mse"].append(mse)
        metrics["rmse"].append(rmse)
        metrics["psnr"].append(psnr)
        metrics["cross_corr"].append(xcorr)
        metrics["mutual_info"].append(mi)
        metrics["f1"].append(f1)
        metrics["recall"].append(recall)

        image_index += 1
```

```
108
109     # --- Calculate Averages ---
110     avg_metrics = {key: float(np.mean(values)) for key,
        values in metrics.items()}
111     hotspot_percentage = (hotspot_images / total_images) *
        100 if total_images > 0 else 0
112     avg_metrics["hotspot_percentage"] = hotspot_percentage
113
114     return avg_metrics
```

Code listing A.3 Model Evaluation Script

A.4 Configuration File – config.json

```
1  {
2      "MODEL_PATH": "models/final/heat_map_model.keras",
3      "DATA_DIR": "C:\\Path\\To\\Dataset\\Directory",
4      "IMAGE_Y": 1024,
5      "IMAGE_X": 1024,
6      "SEQUENCE_LEN": 4,
7      "SEQUENCE_STEP": 5,
8      "FUTURE_STEP": 10,
9      "INDICATORS_COUNT": 3,
10     "BATCH_SIZE": 32,
11     "EPOCHS": 100
12 }
```

Code listing A.4 Configuration File Example

Appendix B

Trained Machine Learning Models

This chapter describes the pre-trained machine learning models included with this thesis.

Model Files Location

All models are included in the directory: `models/final/`

B.1 List of Models

- **heat_map_barcelona.keras** – Model trained on Barcelona data.
- **heat_map_casablanca.keras** – Model trained on Casablanca data.

B.2 Usage Instructions

To use these models for prediction, place the desired model file path in `config.json` or specify it directly in `predict.py`.

Example:

```
"MODEL_PATH": "models/final/heat_map_barcelona.keras"
```

B.3 Notes

Models use different input dimensions and data structure. Barcelona, Casablanca, Chennai models parameters:

- ”IMAGE_Y”: 1024,

- ”IMAGE_X”: 1024,
- ”SEQUENCE_LEN”: 4,
- ”SEQUENCE_STEP”: 5,
- ”FUTURE_STEP”: 10

Bibliography

1. MINER, Mark J; TAYLOR, Robert A; JONES, Cassandra; PHELAN, Patrick E. Efficiency, economics, and the urban heat island. *Environment and Urbanization*. 2017, vol. 29, no. 1, pp. 183–194.
2. RIZWAN, Ahmed Memon; DENNIS, Leung YC; CHUNHO, LIU. A review on the generation, determination and mitigation of Urban Heat Island. *Journal of environmental sciences*. 2008, vol. 20, no. 1, pp. 120–128.
3. SENEVIRATNE, Sonia I; ZHANG, Xuebin; ADNAN, Muhammad; BADI, Wafae; DERECZYNSKI, Claudine; LUCA, A Di; GHOSH, Subimal; ISKANDAR, Iskhaq; KOSSIN, James; LEWIS, Sophie, et al. Weather and climate extreme events in a changing climate. *Cambridge University Press*. 2021.
4. JOHANNSEN, Frederico; SOARES, Pedro MM; LANGENDIJK, Gaby S. On the Deep learning approach for improving the representation of urban climate: the Paris urban heat island and temperature extremes. *Urban Climate*. 2024, vol. 56, p. 102039.
5. JACK, Christopher; PARKER, Craig; KOUAKOU, Yao Etienne; JOUBERT, Bonnie; MCALLISTER, Kimberly A; ILIAS, Maliha; MAIMELA, Gloria; CHERSICH, Matthew; MAKHANYA, Sibusisiwe; LUCHTERS, Stanley, et al. Leveraging data science and machine learning for urban climate adaptation in two major African cities: a HE2AT Center study protocol. *BMJ open*. 2024, vol. 14, no. 6, e077529.
6. O'SHEA, K. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*. 2015.
7. GRAVES, Alex; FERNÁNDEZ, Santiago; SCHMIDHUBER, Jürgen. Multi-dimensional recurrent neural networks. In: *International conference on artificial neural networks*. Springer, 2007, pp. 549–558.

8. GONG, Yuhao; ZHANG, Yuchen; WANG, Fei; LEE, Chi-Han. Deep learning for weather forecasting: A cnn-lstm hybrid model for predicting historical temperature data. *arXiv preprint arXiv:2410.14963*. 2024.

9. CHIN, Seokhyun; LLOYD, Victoria. Predicting climate change using an autoregressive long short-term memory model. *Frontiers in Environmental Science*. 2024, vol. 12, p. 1301343.

10. JUN, Myung-Jin. A comparison of a gradient boosting decision tree, random forests, and artificial neural networks to model urban land use changes: the case of the Seoul metropolitan area. *International Journal of Geographical Information Science*. 2021, vol. 35, no. 11, pp. 2149–2167.

11. BERNHARDSEN, Tor. *Geographic information systems: an introduction*. John Wiley & Sons, 2002.

12. ALCANTARA, CA; ESCOTO, JD; BLANCO, AC; BALOLOY, AB; SANTOS, JA, et al. Geospatial assessment and modeling of urban heat islands in Quezon City, Philippines using ols and geographically weighted regression. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. 2019, vol. 42, pp. 85–92.

13. WANG, Shuai; RAO, Yuhan; CHEN, Jin; LIU, Licong; WANG, Wenqing. Adopting "difference-in-differences" method to monitor crop response to agrometeorological hazards with satellite data: a case study of dry-hot wind. *Remote Sensing*. 2021, vol. 13, no. 3, p. 482.

14. FU, Yingchun; ZHU, Zhe; LIU, Liangyun; ZHAN, Wenfeng; HE, Tao; SHEN, Huanfeng; ZHAO, Jun; LIU, Yongxue; ZHANG, Hongsheng; LIU, Zihan, et al. Remote sensing time series analysis: A review of data and applications. *Journal of Remote Sensing*. 2024, vol. 4, p. 0285.

15. LIN, Li; DI, Liping; ZHANG, Chen; GUO, Liying; ZHAO, Haoteng; ISLAM, Didarul; LI, Hui; LIU, Ziao; MIDDLETON, Gavin. Modeling urban redevelopment: A novel approach using time-series remote sensing data and machine learning. *Geography and Sustainability*. 2024, vol. 5, no. 2, pp. 211–219.

16. AKBARI, H. Potentials of urban heat island mitigation. In: *Proceedings of the International Conference on Passive and Low Energy Cooling for the Built Environment, Santorini, Greece*. 2005, pp. 19–21.

17. UNDERWOOD, B Shane; GUIDO, Zack; GUDIPUDI, Padmini; FEINBERG, Yarden. Increased costs to US pavement infrastructure from future temperature rise. *Nature Climate Change*. 2017, vol. 7, no. 10, pp. 704–707.

18. ODYSSEE-MURE PROJECT. *Energy Efficiency Profile: Czechia*. 2021. Available also from: `https://www.odyssee-mure.eu/publications/efficiency-trends-policies-profiles/czechia.html`. Accessed: 2024-02-19.

19. ING. JIŘÍ TENCAR, Ph.D.; BHATTACHARJEE, Sagnik; CORNEILLE, Manon; DANILCHYK, Tatsiana; VEJRAŽKA, Martin; NAGHDI, Yashar. HODNOCENÍ URBÁNNÍ TEPELNÉ ZRANITELNOSTI ZASTÁVEK HROMADNÉ DOPRAVY. *Ecoten*. 2020. Available also from: `https://adaptacepraha.cz/wp-content/uploads/2020/04/IPR-Prague-Zprava.pdf`.

Contents of Appendices

└── `thesis.pdf` ..... Text of the thesis in PDF format