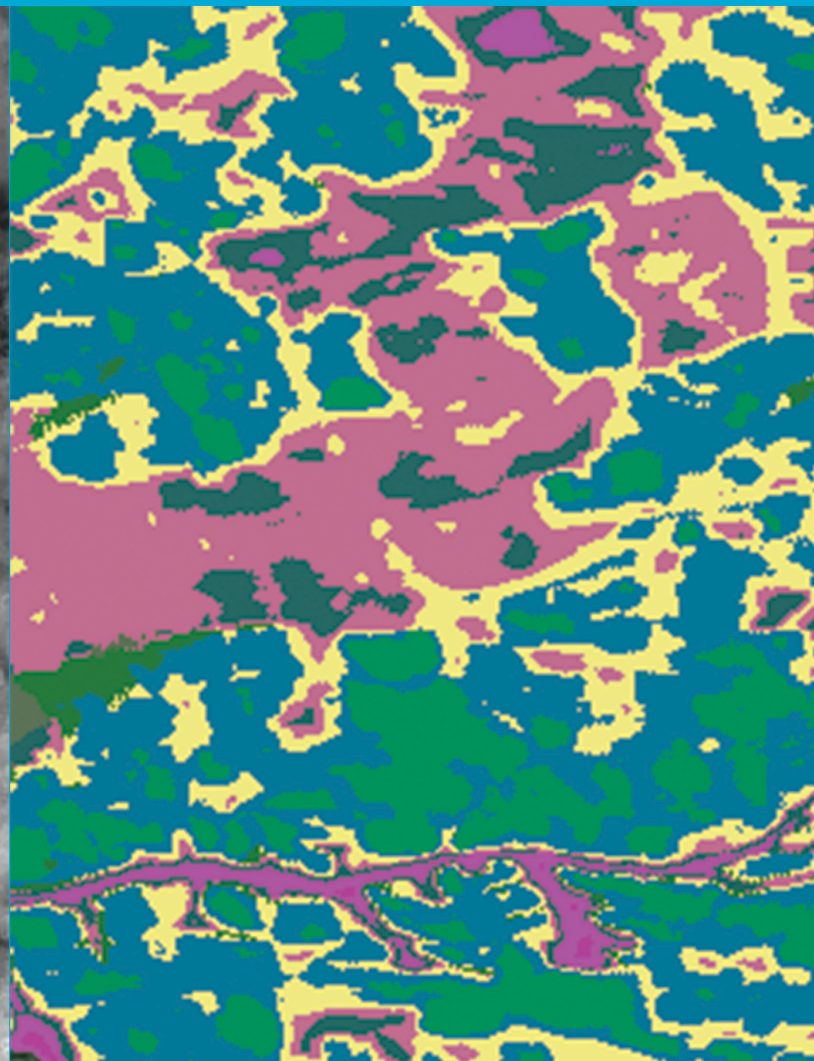
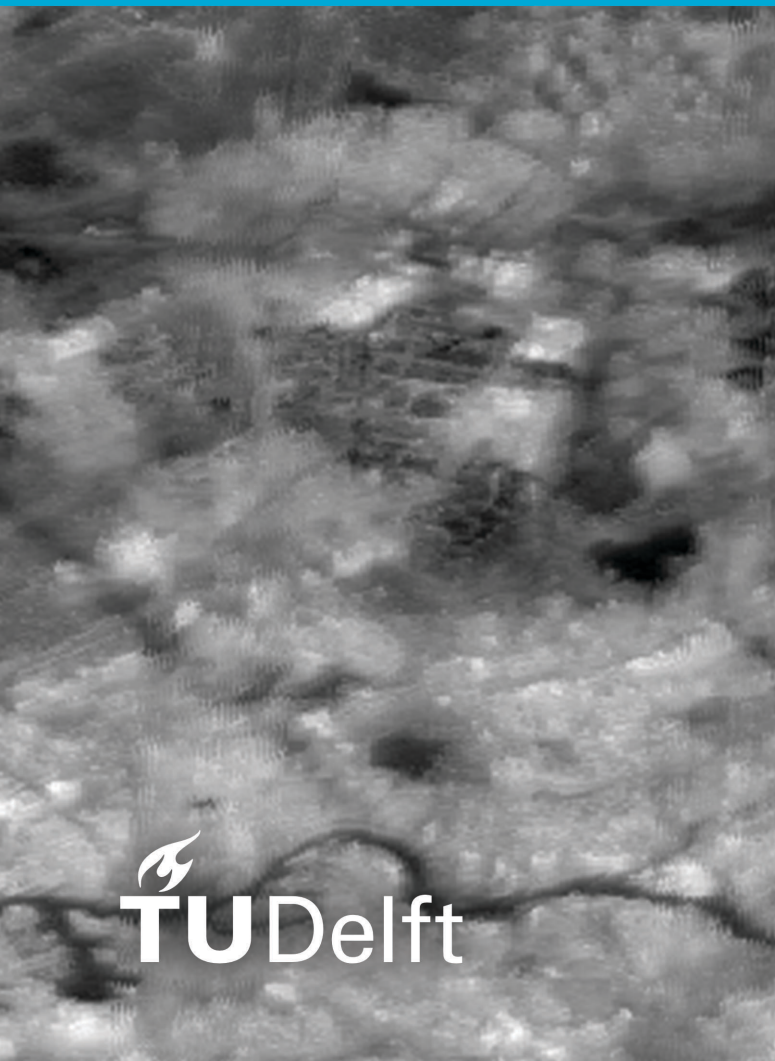


MSc thesis in Geomatics

Urban Local Climate Zone classification through deep learning using spatio-temporal thermal imagery

Michaja van Capel
2024



MSc thesis in Geomatics

**Urban Local Climate Zone classification
through deep learning using
spatio-temporal thermal imagery**

Michaja van Capel

July 2024

A thesis submitted to the Delft University of Technology in
partial fulfillment of the requirements for the degree of Master
of Science in Geomatics

Michaja van Capel: *Urban Local Climate Zone classification through deep learning using spatio-temporal thermal imagery* (2024)

© ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

Supervisors: Dr. Azarakhsh Rafiee
Dr. Roderik Lindenbergh
Co-reader: Dr. Vitali Diaz Mercado

Abstract

Local Climate Zone (LCZ) classification plays a crucial role in understanding and managing urban environments, particularly through the lens of Land Surface Temperature (LST) behavior. This study investigates the effectiveness of a Convolutional Neural Network (CNN) with U-net architecture for classifying urban LCZs using spatio-temporal thermal imagery.

The research addresses several key sub-questions, including the creation of a representable training dataset, optimizing hyperparameters for the U-net model, and assessing the impact of temporal factors on classification performance. An unsupervised clustering approach was adopted to label the training data, utilizing the Iterative Self-Organizing Data Analysis Technique (ISODATA) algorithm on stacks of thermal images to generate clusters based on thermal behavior, which were then manually refined and validated. Through experimentation, optimal hyperparameters were identified: a learning rate of 0.001, a patch size of 64, and the SparseCategoricalCrossentropy loss function. The study also highlights the significant influence of temporal factors, when using daytime and Spring/Summer thermal images for training and testing the model better classification outcomes were obtained compared to nighttime and Autumn/Winter images.

The research contributes to the LCZ classification by incorporating both spatial and temporal dimensions of LST patterns, providing valuable insights for urban planning. The findings demonstrate that a CNN with U-net architecture is highly suitable for classifying urban LCZs, particularly when the dataset captures diverse seasonal and extreme conditions. This approach offers a robust and adaptable framework for urban environmental monitoring and planning. This thesis has explored the utilization of a new source for LCZ classification, providing a useful starting position for further enhancement of standardized LCZ classification. Future work is therefore recommended to focus on integrating additional geospatial data sources, refining classification categories, and integrating the standardized LCZ classification system by Stewart and Oke [2012].

Acknowledgements

First and foremost, I would like to thank my supervisor Dr. Azarakhsh Rafiee who made this work possible. Her guidance, encouragement and feedback carried me through the process of writing my thesis. Also, I would like to thank the other members of my mentor team Dr. Roderik Lindenbergh and Dr. Vitali Diaz for their valuable feedback.

I would also like to give special thanks to Bas, my parents and my friends for their continuous support and involvement throughout the years of my studies. I could not have done it without you.

Contents

1. Introduction	1
1.1. Motivation	1
1.2. Objective and Scope	2
1.3. Research questions	2
1.4. Thesis outline	2
2. Theoretical background and related work	3
2.1. Local Climate Zone Classification	3
2.1.1. The LCZ classification system	3
2.1.2. LCZ and Land Surface Temperature	3
2.1.3. LCZ classification methods	5
2.2. Thermal remote sensing	6
2.3. Unsupervised Clustering methods	7
2.3.1. K-means Clustering	7
2.3.2. ISODATA Clustering	8
2.3.3. Unsupervised clustering methods on time series data	8
2.4. Deep learning architecture	9
2.4.1. Machine learning	9
2.4.2. Deep learning	9
2.4.3. Artificial Neural Networks	9
2.4.4. Convolutional Neural Networks	11
2.4.5. U-net architecture	13
3. Methodology	17
3.1. Data pre-processing	18
3.1.1. Study area and time span selection	18
3.1.2. Data collection	18
3.1.3. Stacking images	19
3.1.4. Data normalization	19
3.2. Training data labeling	19
3.2.1. LCZ-LST analysis	20
3.2.2. Unsupervised clustering methods	20
3.3. Data split	21
3.4. U-Net architecture	21
3.5. Evaluation	22
4. Implementation	25
4.1. Training data preparation	25
4.1.1. LCZ-LST analysis	25
4.2. Training data set	26
4.3. Hyperparameter definition	29
4.3.1. Learning rate	29

Contents

4.3.2. Patch size	30
4.3.3. Loss function	31
5. Results	33
5.1. Full dataset	33
5.1.1. Probability distribution	34
5.2. Seasonal influence	36
5.3. Daytime vs. nighttime	39
5.4. Extreme analysis	42
6. Discussion, conclusion and future work	45
6.1. Discussion	45
6.2. Limitations	45
6.3. Conclusion	46
6.4. Future work	48
A. Reproducibility self-assessment	51
A.1. Marks for each of the criteria	51
A.2. Self-reflection	52
B. Dates and times thermal images	53

List of Figures

2.1. Local Climate Zone system by Stewart and Oke [2012]	4
2.2. Boxplot with LSTs in LCZs for Adana City by Cilek and Cilek [2021]	5
2.3. LST measurements using remote sensing by Singh et al. [2024]	7
2.4. Umbrella of select data science techniques [Choi et al., 2020]	9
2.5. A biological neuron in comparison to an Artificial Neural Network (ANN): (a) Human neuron, (b) Artificial neuron, (c) Biological synapse, (d) ANN synapses [Suzuki, 2013]	10
2.6. Graph of ReLU activation function [Bishop and Bishop, 2023]	11
2.7. A basic architecture example of a CNN [Gu et al., 2019]	12
2.8. Example of a convolutional filter with input image of size 7×7 and a filter kernel of size 3×3 [Baskin et al., 2017]	12
2.9. Example of max pooling and average pooling operations [Hossain and Sajib, 2019]	13
2.10. Example of a fully connected layer [Hossain and Sajib, 2019]	13
2.11. U-net architecture by [Ronneberger et al., 2015]	14
3.1. Workflow	17
3.2. Study area	18
3.3. Drawn polygons on LCZ map by Demuzere et al. [2019]	20
4.1. box plot of LST per LCZ (2022)	25
4.2. box plot of LST per LCZ (2022) zoomed in	25
4.3. ISODATA clustering results	27
4.4. Thermal signatures of resulting clusters from clustering algorithm	28
4.5. Model's performance according to loss and accuracy with different learning rates	29
4.6. Model's performance according to loss and accuracy with different patch sizes	30
4.7. Model's performance according to loss and accuracy with different loss functions	31
5.1. Training and testing with full dataset results	33
5.2. Probability distribution per class for the test data	34
5.3. Average maximum probability per test data pixel	35
5.4. Training and testing with data from Spring/Summer results	36
5.5. Training and testing with data from Autumn/Winter results	36
5.6. Actual mask and classification result after training the model with Autumn/Winter images	38
5.7. Training and testing with a selection of 7 images from Spring/Summer results	39
5.8. Training and testing with data from daytime results	40
5.9. Training and testing with data from nighttime results	40
5.10. Actual mask and classification result after training the model with nighttime images	41

List of Figures

5.11. Thermal signatures peaks	42
A.1. Reproducibility criteria	51

List of Tables

4.1.	LST mean, range, sample variance and standard deviation per LCZ	26
4.2.	Class descriptions based on aerial imagery	27
4.3.	Class representation	28
4.4.	Test accuracy values for different learning rates	30
4.5.	Test accuracy values for different patch sizes	31
5.1.	Test F1 score per class	34
5.2.	Test F1 score per class for Spring/Summer and Autumn/Winter	37
5.3.	Number of images and test accuracy for different selections	38
5.4.	Test F1 score per class for Spring/Summer and Autumn/Winter, using 7 images	39
5.5.	Test F1 score per class for daytime and nighttime	41
5.6.	Number of images and test accuracy for different selections	42
5.7.	Test accuracy values for different image selections	43
A.1.	Evaluation of reproducibility criteria	51
B.1.	Dates and times of selected thermal images	54

Acronyms

AI	Artificial Intelligence	9
ANN	Artificial Neural Network	xi
AppEARS	Application for Extracting and Exploring Analysis Ready Samples	18
ASTER	Advanced Spaceborne Thermal Emission and Reflection Radiometer	7
CNN	Convolutional Neural Network	v
DL	Deep Learning	1
ECOSTRESS	ECOSystem Spaceborne Thermal Radiometer Experiment on Space Station	7
FCN	Fully Convolutional Network	13
FN	false negatives	22
FP	false positives	22
GIS	Geographical Information System	6
ISODATA	Iterative Self-Organizing Data Analysis Technique	v
ISS	International Space Station	7
LCZ	Local Climate Zone	v
LiDAR	Light Detection and Ranging	5
LST	Land Surface Temperature	v
LWIR	long-wavelength infrared	6
ML	Machine Learning	7
MODIS	Moderate Resolution Imaging Spectroradiometer	7
NASA	National Aeronautics and Space Administration	7
OA	overall accuracy	22
ReLU	Rectified Linear Unit	11
SVF	Sky View Factor	6
TP	true positives	22
WUDAPT	World Urban Database and Access Portal Tools	6

1. Introduction

In recent decades, urbanization has been rapidly transforming the global landscape, with a significant proportion of the world's population now living in urban areas [Zhang et al., 2022]. This urban growth has given rise to numerous challenges, including the negative effects on urban micro-climates and the well-being of urban inhabitants. Understanding and characterizing the urban climate is crucial for mitigating these challenges and creating sustainable urban environments [Ren et al., 2022]. To this end, the concept of LCZs has emerged as a valuable framework for urban climate classification and analysis [Stewart and Oke, 2012].

Recent advancements in remote sensing technologies, together with the power and development of Deep Learning (DL) algorithms, has provided promising opportunities for automated and data-driven LCZ classification. Among the various remote sensing modalities, thermal imagery stands out as a rich source of information for capturing the spatio-temporal thermal dynamics of urban areas. Thermal imagery enables the observation of thermal patterns within different LCZs. To harness the potential of thermal imagery for LCZ classification, this research aims to explore the suitability of the CNN architecture, specifically the U-net model. In this approach, stacks of thermal images will serve as input data, enabling the model to effectively capture temporal relations while maintaining spatial coherence. In previous LCZ-LST studies only one snapshot, or diurnal, seasonal and annual patterns were analyzed [Liu et al., 2018]. This thesis aims to classify urban LCZs based on underlying LST patterns.

Incorporating temporal thermal heat imagery, and taking spatial relations into account, LCZ classification methods can potentially achieve higher accuracy and objectivity in defining LCZs. Where thermal heat imagery can capture fine-scale temperature variations within an urban area [Zhao et al., 2021], LCZ classification methods based on multi-spectral satellite imagery may not capture the micro-climate variations accurately. The aim of this thesis is to optimize an urban LCZ classification using temporal thermal imagery.

1.1. Motivation

As mentioned before, the rapid urbanization experienced globally has significantly impacted urban climates, influencing both environmental conditions and human well-being [Ren et al., 2022]. The need for accurate and efficient methods to classify and analyze urban climates is pressing. The concept of LCZs offers a structured framework for this purpose, yet the accuracy and scalability of LCZ mapping remain challenging. The integration of advanced remote sensing technologies and deep learning algorithms, particularly the use of thermal imagery, presents a novel approach to address these challenges. This research is motivated by the potential to improve urban climate management and contribute to sustainable urban development through enhanced LCZ classification methods.

1.2. Objective and Scope

The primary objective of this thesis is to develop and optimize an urban LCZ classification method using temporal thermal imagery with a CNN with U-net architecture. The scope of this research includes:

- Collecting and pre-processing a representative dataset of thermal images, together with their manually created ground truth labels,
- Designing and training a U-net based CNN for LCZ classification,
- Evaluating the model's performance in different temporal contexts (day-night, seasonal).

The goal of the thesis is to gain insight in enhancing the process of LCZ classification, by exploring a new source. The ultimate goal is to create a versatile LCZ classification framework that can be applied to various urban settings globally, providing planners and policymakers with reliable tools for urban climate management.

1.3. Research questions

The main research question for this thesis is: *To what extent is a CNN with U-net architecture using spatio-temporal thermal imagery suitable for the classification of urban LCZs?*

In order to assess the performance of the U-net network, the different contributing factors need to be considered. The training data and architecture of the deep learning network can significantly influence the accuracy of the model. Therefore the following research sub-questions need to be addressed to provide an exploration of the main research question:

- How can a representable training data set be collected?
- When it comes to the architecture of U-net, what values for the hyperparameters of the deep learning network lead to the best classification result?
- What is the impact of temporal frequency (day-night, seasonal) on the classification performance?

1.4. Thesis outline

The thesis consists of 6 chapters. In Chapter 2 a literature review is provided. This chapter provides a comprehensive review of existing literature on LCZ classification, remote sensing technologies and the necessary DL theory. Chapter 3 details the steps of the methodology. In Chapter 4, a technical implementation of the methodology is provided. In Chapter 5, the results of the different experiments are presented. Chapter 6 the key findings are summarized, together with the implications of the findings, potential limitations, and areas for future research.

2. Theoretical background and related work

2.1. Local Climate Zone Classification

2.1.1. The LCZ classification system

In 2012, the LCZ classification system was introduced by Stewart and Oke [2012]. Climate classifications were typically formulated to describe climate zones at larger scales, making them ineffective when applied to smaller, micro-scale areas. Sites in cities with very different physical and climatological features were usually described only as “urban” or “rural”. The LCZ system aims to overcome this [Aslam and Rana, 2022]. Within this system, there are 17 distinct zones, each characterized by its unique combination of surface structure, cover, and human activity. By considering these factors, the LCZ system provides a more accurate and detailed representation of the climate within specific areas. The different LCZs and their definition are shown in Figure 2.1. LCZ 1-10 are different built-up classes, and LCZ A-G different land cover types [Stewart and Oke, 2012].

2.1.2. LCZ and Land Surface Temperature

LST represents the radiant skin temperature of the Earth’s land surface, as determined by the absorption, reflection and emission of solar radiation [Khan et al., 2021].

The relationship between LCZs and LST is that the physical characteristics of each LCZ type influence the LST within that zone. The composition and arrangement of land cover types directly influence the distribution of energy absorbed and emitted at the land surface, thereby establishing a correlation with LST variations. Understanding these correlations has important potential for managing urban heat islands, improving urban microclimates, addressing climate change impacts, and assessing environmental and ecological aspects of urban areas, and has therefore been the subject of numerous studies. The research by Cilek and Cilek [2021] shows differences in the mean LST values and differences per LCZ in Adana City in Turkey. Figure 2.2 shows a box plot of the different LST values per LCZ on the 13th of August in Adana City. For example: LCZs with high proportions of impervious surfaces, such as urban centers or compact high-rise areas, tend to have higher LSTs due to the absorption and retention of solar radiation by buildings and pavement. LCZs with more vegetation, such as parks or forests, generally have lower LSTs due to the cooling effect of vegetation through evapotranspiration and shading. Some LCZs do not show considerable differences in LST values when using one thermal image [Lottian et al., 2019]. Another study by [Zhao et al., 2021] shows differences in seasonal LST variabilities per LCZ. A general trend found is that the diurnal LST variation increases with the urbanization index Chen et al. [2017].

2. Theoretical background and related work

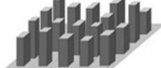









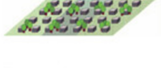

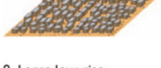




Built types	Definition	Land cover types	Definition
 <p>1. Compact high-rise</p>	Dense mix of tall buildings to tens of stories. Few or no trees. Land cover mostly paved. Concrete, steel, stone, and glass construction materials.	 <p>A. Dense trees</p>	Heavily wooded landscape of deciduous and/or evergreen trees. Land cover mostly pervious (low plants). Zone function is natural forest, tree cultivation, or urban park.
 <p>2. Compact midrise</p>	Dense mix of midrise buildings (3–9 stories). Few or no trees. Land cover mostly paved. Stone, brick, tile, and concrete construction materials.	 <p>B. Scattered trees</p>	Lightly wooded landscape of deciduous and/or evergreen trees. Land cover mostly pervious (low plants). Zone function is natural forest, tree cultivation, or urban park.
 <p>3. Compact low-rise</p>	Dense mix of low-rise buildings (1–3 stories). Few or no trees. Land cover mostly paved. Stone, brick, tile, and concrete construction materials.	 <p>C. Bush, scrub</p>	Open arrangement of bushes, shrubs, and short, woody trees. Land cover mostly pervious (bare soil or sand). Zone function is natural scrubland or agriculture.
 <p>4. Open high-rise</p>	Open arrangement of tall buildings to tens of stories. Abundance of pervious land cover (low plants, scattered trees). Concrete, steel, stone, and glass construction materials.	 <p>D. Low plants</p>	Featureless landscape of grass or herbaceous plants/crops. Few or no trees. Zone function is natural grassland, agriculture, or urban park.
 <p>5. Open midrise</p>	Open arrangement of midrise buildings (3–9 stories). Abundance of pervious land cover (low plants, scattered trees). Concrete, steel, stone, and glass construction materials.	 <p>E. Bare rock or paved</p>	Featureless landscape of rock or paved cover. Few or no trees or plants. Zone function is natural desert (rock) or urban transportation.
 <p>6. Open low-rise</p>	Open arrangement of low-rise buildings (1–3 stories). Abundance of pervious land cover (low plants, scattered trees). Wood, brick, stone, tile, and concrete construction materials.	 <p>F. Bare soil or sand</p>	Featureless landscape of soil or sand cover. Few or no trees or plants. Zone function is natural desert or agriculture.
 <p>7. Lightweight low-rise</p>	Dense mix of single-story buildings. Few or no trees. Land cover mostly hard-packed. Lightweight construction materials (e.g., wood, thatch, corrugated metal).	 <p>G. Water</p>	Large, open water bodies such as seas and lakes, or small bodies such as rivers, reservoirs, and lagoons.
 <p>8. Large low-rise</p>	Open arrangement of large low-rise buildings (1–3 stories). Few or no trees. Land cover mostly paved. Steel, concrete, metal, and stone construction materials.	VARIABLE LAND COVER PROPERTIES	
 <p>9. Sparsely built</p>	Sparse arrangement of small or medium-sized buildings in a natural setting. Abundance of pervious land cover (low plants, scattered trees).	b. bare trees	Leafless deciduous trees (e.g., winter). Increased sky view factor. Reduced albedo.
 <p>10. Heavy industry</p>	Low-rise and midrise industrial structures (towers, tanks, stacks). Few or no trees. Land cover mostly paved or hard-packed. Metal, steel, and concrete construction materials.	s. snow cover	Snow cover >10 cm in depth. Low admittance. High albedo.
		d. dry ground	Parched soil. Low admittance. Large Bowen ratio. Increased albedo.
		w. wet ground	Waterlogged soil. High admittance. Small Bowen ratio. Reduced albedo.

Figure 2.1.: Local Climate Zone system by Stewart and Oke [2012]

Nevertheless, despite the demonstrated differences between LCZs and the observed thermal patterns, all these studies have concluded that more information and detailed investigations are needed to eliminate the gap between the LCZ and LST relationships. This gap arises due to several factors. Firstly, some LCZs do not show considerable differences in LST values, highlighting the complexity of thermal behavior within and between different urban forms. Secondly, studies are often limited to one city, which restricts the generalizability of the findings across different urban contexts. Lastly, many studies rely on one or a few thermal images, which may not capture the full variability of temperature patterns over time.

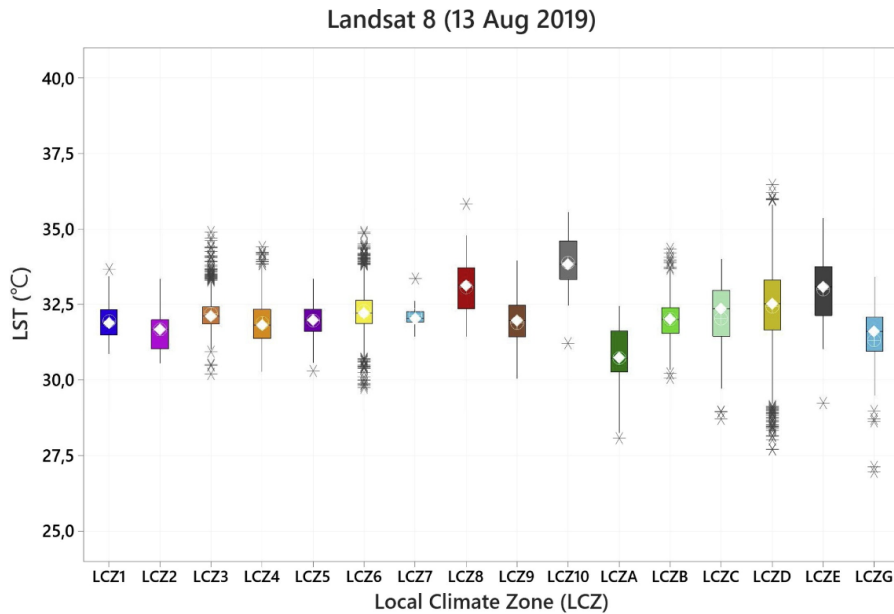


Figure 2.2.: Boxplot with LSTs in LCZs for Adana City by Cilek and Cilek [2021]

2.1.3. LCZ classification methods

The methods used in literature to create **LCZ** classifications can be summarized into two main categories:

- Manual sampling,
- Geospatial data analysis.

Manual sampling involves field surveys, where direct observation and measurement of **LCZs** are conducted in situ. It also includes expert judgment, where classification is based on the expertise and manual interpretation of local urban landscapes. Manual sampling is time-consuming and can be prone to biased results due to variations between different surveys. Geospatial data analysis techniques utilize geospatial data in various formats to classify **LCZs**. These techniques include vector data analysis, which involves using shapefiles and boundary data, and raster data analysis, which employs gridded datasets such as satellite imagery. Also, a combination of both vector and raster data can be used in spatial analysis to comprehensively examine urban morphology. Multiple geospatial data sources in different formats are used in the literature to create **LCZ** classifications. A few important data sources include:

- Multi-spectral satellite imagery,
- Ground-level imagery,
- Aerial imagery,
- Light Detection and Ranging (**LiDAR**) data,
- Derived geospatial data.

2. Theoretical background and related work

Derived geospatial data refers to urban planning information generated from other geospatial data sources. For instance, the parameters building height, mean street width and building surface fraction can be derived from a LiDAR point cloud dataset.

An example of using ground-level imagery for LCZ classification is performed by Xu et al. [2019]. A CNN is trained with ground-level images from Google Street View. The utilization of different data sources for LCZ classification offers unique advantages and limitations. Several LCZ classification studies also utilize a combination of geospatial data sources. Spatial analysis techniques with Geographical Information System (GIS) software can be used to integrate multiple datasets. Utilizing a combination of data sources along with derived geospatial data using GIS software is more data-intensive than other methods, but it can take into account numerous characteristics of LCZs. For example, Cilek and Cilek [2021] created a LCZ classification in Adana city, Turkey using five parameters including building height, building surface fraction, aspect ratio, pervious and impervious surface fraction. Using the parameters, a vector-based analysis was performed using GIS software and the LCZs were selected based on a decision tree. Another study by Zheng et al. [2018] created a LCZ classification of Hong Kong with the same parameters and method, but also using areal mean Sky View Factor (SVF) of non-building areas of the sample site and mean street width.

At larger scales, classification using multi-spectral satellite imagery can be used. Demuzere et al. [2019] mapped Europe into LCZs using tools and techniques developed as part of the World Urban Database and Access Portal Tools (WUDAPT) project. The Random Forest classifier in Google Earth Engine was used for training, enabling a LCZ classification based on Landsat satellite imagery. LCZ classification using remote sensing is fast, cost-effective and can therefore be applied to larger scales. However, classification approaches based on satellite images have limitations regarding the characterisation of three-dimensional features such as building heights. In summary, while primary data sources provide essential information, the integration of multiple geospatial data sources and derived data enhances the detail and accuracy of LCZ classifications, accommodating various spatial and structural characteristics.

2.2. Thermal remote sensing

Thermal satellite imagery offers a valuable window into the thermal behavior of urban environments [Zhao et al., 2021]. Thermal images are obtained through specialized sensors onboard Earth-observing satellites. Unlike traditional optical imagery, that uses visible light, thermal sensors detect long-wavelength infrared (LWIR) radiation (8-14 μm) emitted by objects based on their temperature. This range of the electromagnetic spectrum is significant because it is where the Earth's surface and atmosphere emit most of their thermal energy. The surface emissivity measurements are translated and converted to LST, taking into account the atmospheric properties, zenith and azimuth angle of the satellite. The temperature values enable the depiction of thermal contrasts, allowing to distinguish temperature gradients within urban areas [Briottet et al., 2016]. Figure 2.3 shows objects in an urban environment with different LWIR values detected by a thermal sensor.

Thermal remote sensing has a wide range of applications. It is used to map land and ocean surface temperatures, which is essential for studying climate change, weather patterns, and ocean currents. In urban areas, it helps analyze temperature variations to understand the

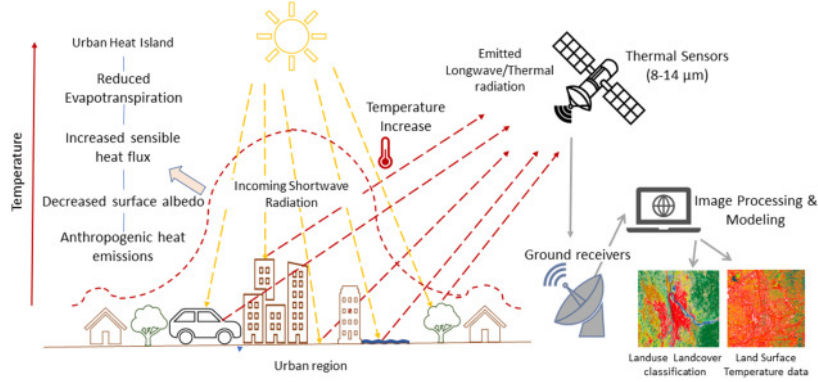


Figure 2.3.: LST measurements using remote sensing by Singh et al. [2024]

effects of human activities and improve urban planning. Environmental monitoring applications include detecting thermal anomalies to monitor forest fires, volcanic activity, and other environmental changes. For agriculture, it is used to assess crop health, soil moisture levels, and water stress by examining temperature variations in fields [Prakash, 2000].

Satellites like Landsat, Moderate Resolution Imaging Spectroradiometer (MODIS), Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER), and ECOSystem Spaceborne Thermal Radiometer Experiment on Space Station (ECOSTRESS) provide crucial thermal imagery data. Landsat offers thermal images with a spatial resolution of 120 meters. MODIS, mounted on the Terra and Aqua satellites of National Aeronautics and Space Administration (NASA), provides data with a spatial resolution of 1 kilometer at nadir and high temporal resolution, capturing images every 1-2 days, making it ideal for large-scale environmental monitoring. ASTER, on the other hand, provides high-resolution thermal infrared data with a spatial resolution of 90 meters, useful for detailed geological and environmental studies. ECOSTRESS, mounted on the International Space Station (ISS), provides thermal imagery with a spatial resolution of 70 meters and a high temporal resolution, capturing data at different times of the day due to the orbit of ISS, which is particularly beneficial for monitoring diurnal temperature variations. For this thesis, the thermal imagery of the ECOSTRESS mission will be used, because of its high spatial and temporal resolution.

2.3. Unsupervised Clustering methods

2.3.1. K-means Clustering

K-means clustering stands as a widely adopted unsupervised Machine Learning (ML) technique that subdivides a dataset into k unique, non-overlapping clusters. The objective function of K-means clustering is to minimize the the sum of squared distances from every point within a cluster to its centroid, also referred to as the within-cluster variance, given by Equation 2.1. First a pre-defined number (k) of cluster centroids are randomly initialized. After which every data point is assigned to the nearest centroids. The centroids will then be updated as the mean of the data points that were assigned to each cluster. This will be repeated until convergence occurs [Bishop, 2006].

2. Theoretical background and related work

$$\sum_{i=1}^k \sum_{x \in C_i} \|x - \mu_i\|^2 \quad (2.1)$$

where:

- k is the number of clusters,
- C_i is the i -th cluster,
- μ_i is the centroid of the i -th cluster, and
- $\|x - \mu_i\|^2$ is the squared Euclidean distance between data point x and centroid μ_i .

2.3.2. ISODATA Clustering

ISODATA is a variant of the K-means algorithm, but it dynamically adjusts the number of clusters based on the distribution of data points. It is useful when the number of clusters is unknown or fluctuates over time. **ISODATA** does not have a specific objective function like K-means clustering, as the number of clusters and cluster parameters are dynamically adjusted during the iterative process.

Initially, a starting number of clusters k is initialized, together with a merging threshold and splitting threshold. The merging threshold defines the maximum allowable distance between cluster centroids for clusters to be merged. If the distance between any two cluster centroids is less than this threshold, the clusters are combined. This helps in reducing the number of clusters when they are too close to each other, ensuring well-separated clusters. The splitting threshold determines the maximum allowable variance within a cluster. If a cluster's variance exceeds this threshold, the cluster is split into two smaller clusters. Splitting helps to break down large, heterogeneous clusters into smaller, more homogeneous clusters, improving clustering quality. During the same iterative process as for K-means clustering, clusters are repeatedly merged and split based on these thresholds until a stable clustering configuration is achieved [Ball and Hall, 1965].

2.3.3. Unsupervised clustering methods on time series data

With an increase in temporal data mining research, there has been expanding interest in clustering time series. Clustering time series comes with challenges because of temporal dependencies and potential high dimensionality [Aghabozorgi et al., 2015]. Despite the challenges, unsupervised clustering methods like K-means and **ISODATA** can be applied to timeseries data. When these methods are used, the resulting cluster centroids are themselves timeseries, with each centroid providing a representative value for each time step.

When it comes to time series data of thermal satellite imagery, Liu et al. [2018] applied spatio-temporal clustering using K-means to thermal imagery, effectively characterizing 17 homogeneous geographical clusters with thermal patterns over time.

2.4. Deep learning architecture

2.4.1. Machine learning

Artificial Intelligence (AI) represents a scientific advancement wherein machines are developed to execute tasks that align with human intelligence, including teaching, reasoning and self-correction. The primary objective of AI is to delegate human-like responsibilities to computers [Aggarwal et al., 2022]. ML is a domain dedicated to the learning aspect of AI through the creation of algorithms that effectively model a given dataset [Choi et al., 2020]. The three terms AI, ML and DL are frequently used interchangeably, despite their different natures. Figure 2.4 shows the relationship between the different terms, where ML is a narrower area of study within AI.

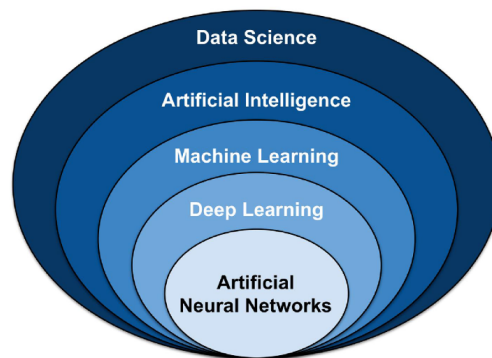


Figure 2.4.: Umbrella of select data science techniques [Choi et al., 2020]

2.4.2. Deep learning

As shown in Figure 2.4, DL is a branch of ML. DL represents a transformative approach to ML that aims to develop a model that matches the level of the human brain in solving complex problems. The human brain consists of a group of neurons. DL tries to mimic the structure and function of the human neural connections through ANN techniques. An ANN with multiple layers is capable of learning hierarchical representations of large datasets, where it autonomously learns to extract intricate patterns and features, enabling to make accurate predictions, classifications or decisions on new, unseen data [Suzuki, 2013]. DL has revolutionized numerous domains, ranging from computer vision to natural language processing [Aggarwal et al., 2022]. Figure 2.5 shows a human neuron and synapse in comparison to an artificial neuron and synapse. Section 2.4.3 will provide a more detailed exploration of ANNs.

2.4.3. Artificial Neural Networks

ANNs are the fundamental building blocks of DL. They are computational models composed of interconnected nodes (neurons) organized in layers: an input layer, a variable amount of hidden layers, and an output layer. Each neuron performs simple mathematical operations

2. Theoretical background and related work

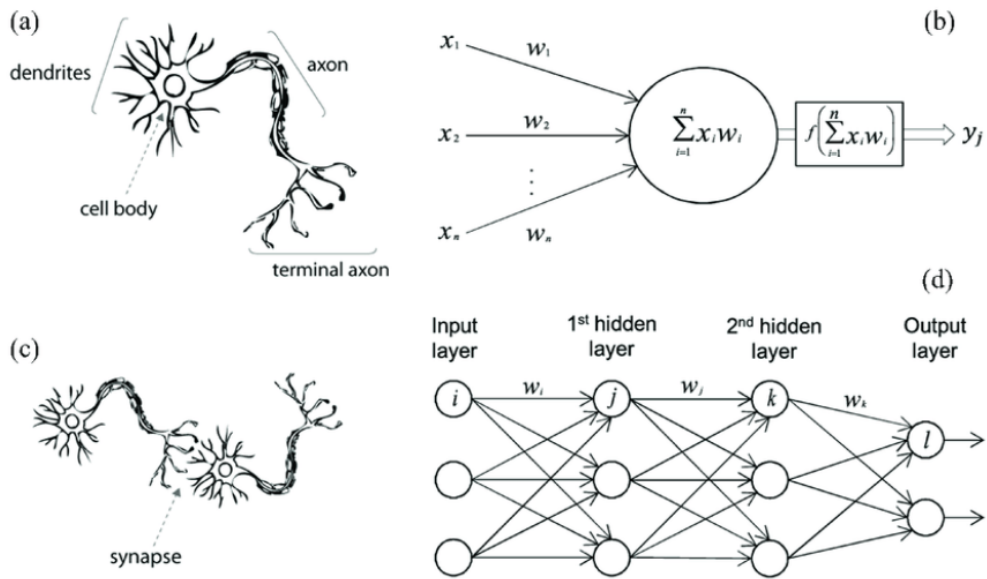


Figure 2.5.: A biological neuron in comparison to an ANN: (a) Human neuron, (b) Artificial neuron, (c) Biological synapse, (d) ANN synapses [Suzuki, 2013]

on its inputs and passes the result to the next layer. The output of the neuron y is given from processing a in a non-linear activation function $f(\cdot)$. There are different activation functions used in ANN studies [Bishop and Bishop, 2023].

$$y = f(a) \quad (2.2)$$

Where a is given by Equation 2.3. x_i is a set of input variables, which are multiplied by parameters w_i , called weights. The weights are comparable to the synaptic strength of a biological neural network. All the weighted input signals are summed. The offset parameter w_0 , also known as bias, is added to this result [Bishop and Bishop, 2023].

$$a = \sum_{i=1}^M w_i x_i + w_0 \quad (2.3)$$

Through a process called backpropagation, ANNs adjust their internal parameters (weights and biases) iteratively during training to minimize the discrepancy between their predictions and the actual targets in the training data. This iterative learning process allows ANNs to learn complex relationships and representations within the data, facilitating tasks such as classification, regression, and pattern recognition [Suzuki, 2013]. In computer vision, an image is an input for an ANN, with its pixels considered as input neurons, fully connected to a series of hidden layers. The output layer exists of neurons indicating the probability of the image or every pixel belonging to a possible class.

Activation function

As previously explained, there are different nonlinear activation functions used in ANN studies. The activation function activates nodes that meet certain criteria, and de-activates nodes that do not. This ensures that the model can focus on “useful” nodes that will be further used in the deeper layers of the network. One of the most commonly used activation functions is the Rectified Linear Unit (ReLU), because of its computational efficiency and fast convergence. The function returns zero if it receives any negative input, but for any positive value x it returns that value back. The graph of the ReLU activation function is shown in Figure 2.6.

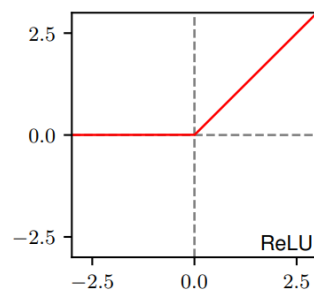


Figure 2.6.: Graph of ReLU activation function [Bishop and Bishop, 2023]

2.4.4. Convolutional Neural Networks

A CNN is a special type of an ANN that has proven to be efficient for processing and analysing visual data, such as images and videos [Ajit et al., 2020]. Traditional ANNs consist of interconnected layers of neurons, where each neuron is connected to every neuron in the adjacent layers. For image data, each pixel is considered as an input neuron with its own input parameters. The challenge that comes with this, is that the number of trainable parameters grows quickly when using image datasets with multiple bands. This complexity is highly limited to computational power. Additionally, the spatial context of the attributes in the image is overlooked. In an image, nearby pixels likely share stronger correlations than those situated far apart. However, a traditional ANN does not take this into account [Choi et al., 2020].

CNNs overcome this challenge by maintaining the spatial coherence among pixels in an image. In mathematics, particularly in functional analysis, a convolution is an operation on two functions that produces a third function, representing the amount of overlap between the two original functions as one is shifted over the other. This operation is widely used in signal processing, probability theory, and various branches of engineering and mathematics [Bracewell, 1999]. In CNNs, nodes in one layer are only connected to a subset of nodes in the previous layer, instead of all the nodes. This grid-like data processing of CNNs is made possible with the presence of convolutional layers, pooling layers, and fully connected layers [Hossain and Sajib, 2019]. A basic architecture example of a CNN is shown in Figure 2.7. The different layers will be further explained in the following sections.

2. Theoretical background and related work

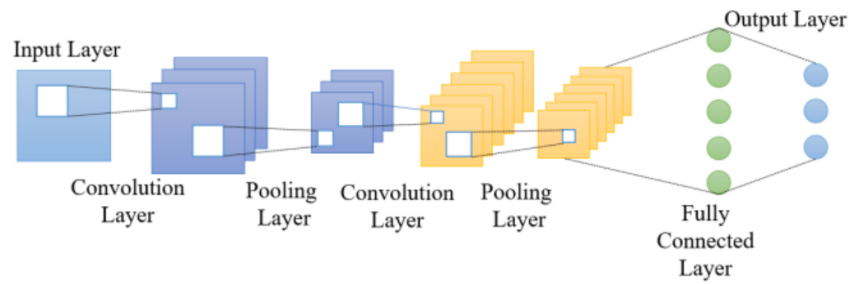


Figure 2.7.: A basic architecture example of a CNN [Gu et al., 2019]

Convolutional layer

Convolutional layers apply filters to input data to extract features. Instead of inputting individual pixels, CNNs supply image patches to specific nodes in the next layer of nodes, conserving the spatial context. These patches of nodes are called convolutional filters. These filters enable CNNs to leverage the inherent structure of image data, leading to a reduction in calculations and enhancement of feature extraction capabilities [Choi et al., 2020]. Convolutions find extensive application in image processing, serving purposes like image blurring, sharpening and edge detection [Bradski and Kaehler, 2008]. An example of a convolutional filter is shown in Figure 2.8.

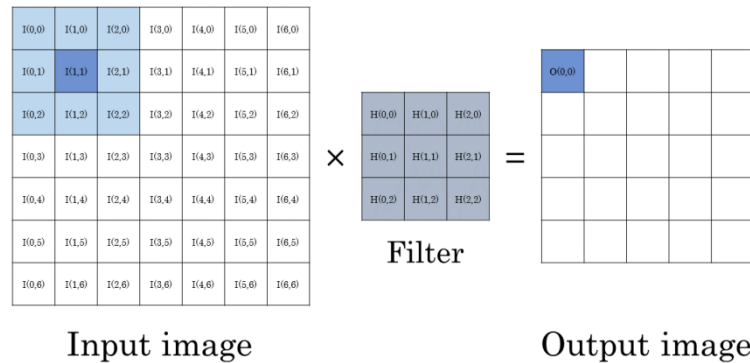


Figure 2.8.: Example of a convolutional filter with input image of size 7×7 and a filter kernel of size 3×3 [Baskin et al., 2017]

Pooling layer

Following a convolutional layer, pooling layers downsample the feature maps to reduce dimensionality. The number of parameters to be computed is reduced by a pooling layer, but the important features are still present [O'Shea and Nash, 2015]. Pooling layers can also prevent overfitting of the model [Hossain and Sajib, 2019]. Different pooling operations can be applied: maximum, sum or average pooling. Figure 2.10 shows an example of a max and average pooling operation.

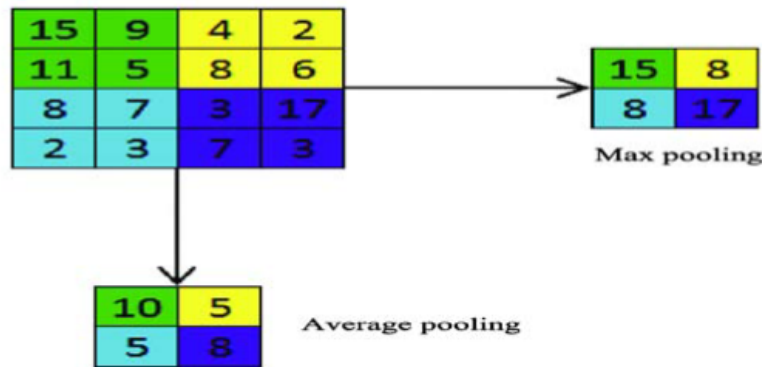


Figure 2.9.: Example of max pooling and average pooling operations [Hossain and Sajib, 2019]

Fully connected layer

Following a pooling layer, there is often a fully connected layer found at the end of the architecture of a CNN. Fully connected layers integrate the extracted features by convolutions and pooling layers and produce class scores for classification or regression tasks. This is the same behaviour as layers in standard ANNs [Hossain and Sajib, 2019].

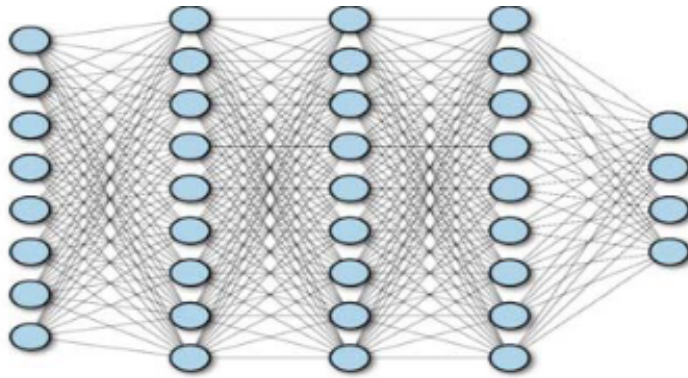


Figure 2.10.: Example of a fully connected layer [Hossain and Sajib, 2019]

2.4.5. U-net architecture

For this thesis, a U-net architecture will be used. The U-net architecture is a variant of a CNN, based on a Fully Convolutional Network (FCN) specifically. The FCN is an architecture developed by Long et al. [2014] that outputs a pixel-wise class labels of the input image, called semantic segmentation. This is achieved by replacing fully connected layers at the end with convolutional layers, and by incorporating upsampling layers, creating an output with the same pixel density as the input image [Garcia-Garcia et al., 2017].

Building upon the FCN architecture, the U-net was developed by Ronneberger et al. [2015] for biomedical image segmentation. The U-net modifies and extends the FCN to enable working

2. Theoretical background and related work

with fewer training images and to yield more precise semantic segmentations. The architecture's name, U-net, is derived from its distinctive U-shaped structure. The model consists of two symmetric paths, being the arms of the U shape. The left arm is the contracting path that captures context, where the right arm is the expansive path that upsamples and enables precise localization. The U-net architecture developed by [Ronneberger et al. \[2015\]](#) is shown in [Figure 2.11](#).

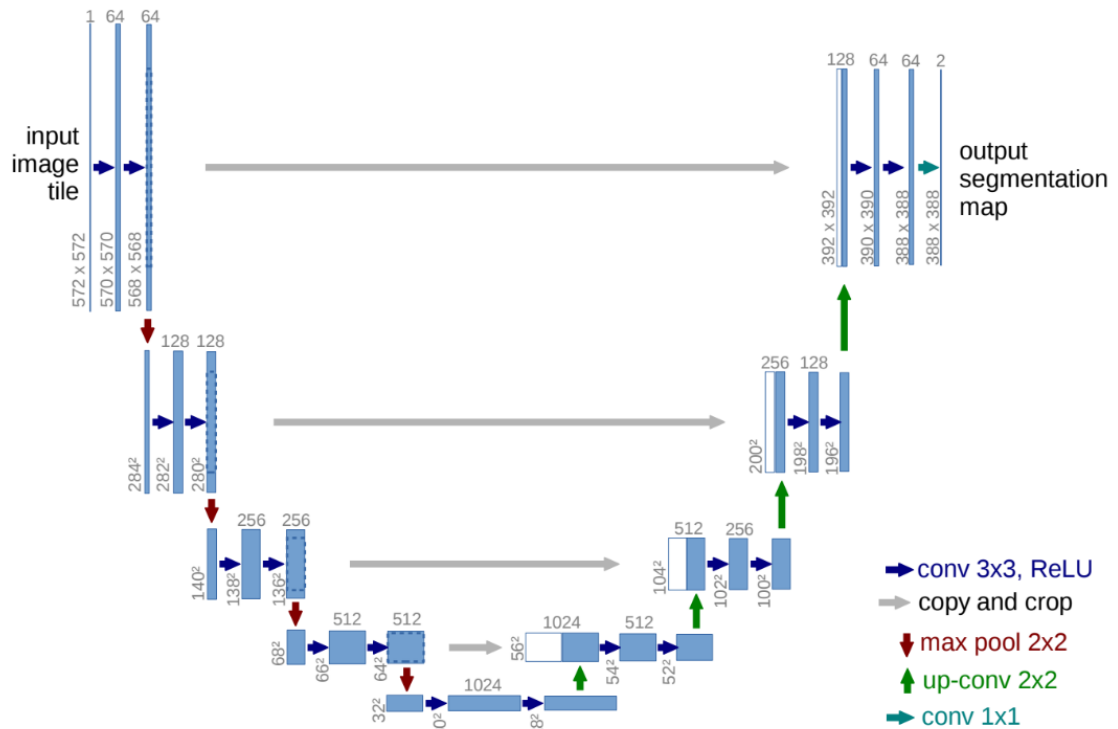


Figure 2.11.: U-net architecture by [[Ronneberger et al., 2015](#)]

The contracting path, also called the encoder, operates similarly to traditional CNNs. It involves repeated applications of two 3×3 convolutions (without padding), each followed by a ReLU and a 2×2 max pooling operation with stride 2 for downsampling. At each downsampling step, the number of feature channels doubles, allowing the network to capture more complex features while reducing the spatial dimensions of the input. At the bottom part of the U, known as the bottleneck, the architecture includes two convolutional layers followed by ReLU activations. At this location, the image size is at its smallest, and the number of feature channels is at its highest. The expansive part, also called the decoder path, is designed to upsample the feature maps and restore the original image size. Each step in this path involves upsampling the feature map using a 2×2 transposed convolution (up-convolution) which halves the number of feature channels. This upsampled feature map is then concatenated with the correspondingly cropped feature map from the encoder path, which helps to retain high-resolution features. Following the concatenation, two 3×3 convolutional layers with ReLU activations are applied. These skip connections are essential for preserving fine-grained details and ensuring precise localization. In the final layer, a 1×1 convolution is used to map each 64-component feature vector to the desired number of output classes.

2.4. Deep learning architecture

This produces an output feature map that is the same size as the input image, with each pixel containing the classification scores.

Overall, the U-Net architecture excels in combining contextual information from the encoder with high-resolution features from the decoder, making it effective for precise semantic segmentation tasks for images.

3. Methodology

This chapter provides an overview of the steps involved in the research for this thesis. A summary of the steps is illustrated in Figure 3.1. In this chapter, the purpose of the steps will be explained. The details on the implementation will be provided in Chapter 4.

The methodology of this thesis can be divided into three main parts, namely (1) data collection and pre-processing, (2) training and application of the CNN model, and (3) the evaluation of the results. The first part starts with the definition of a study area and time span. For this study area and time span, thermal satellite data is collected and labeled data is created. With this data, several pre-processing steps are necessary to prepare the data in a way that it can be used by the CNN. The CNN with U-Net architecture is employed for training the classification of LCZs. The model is optimized through evaluation and analysis of the results, involving adjustments to the training data and hyperparameters.

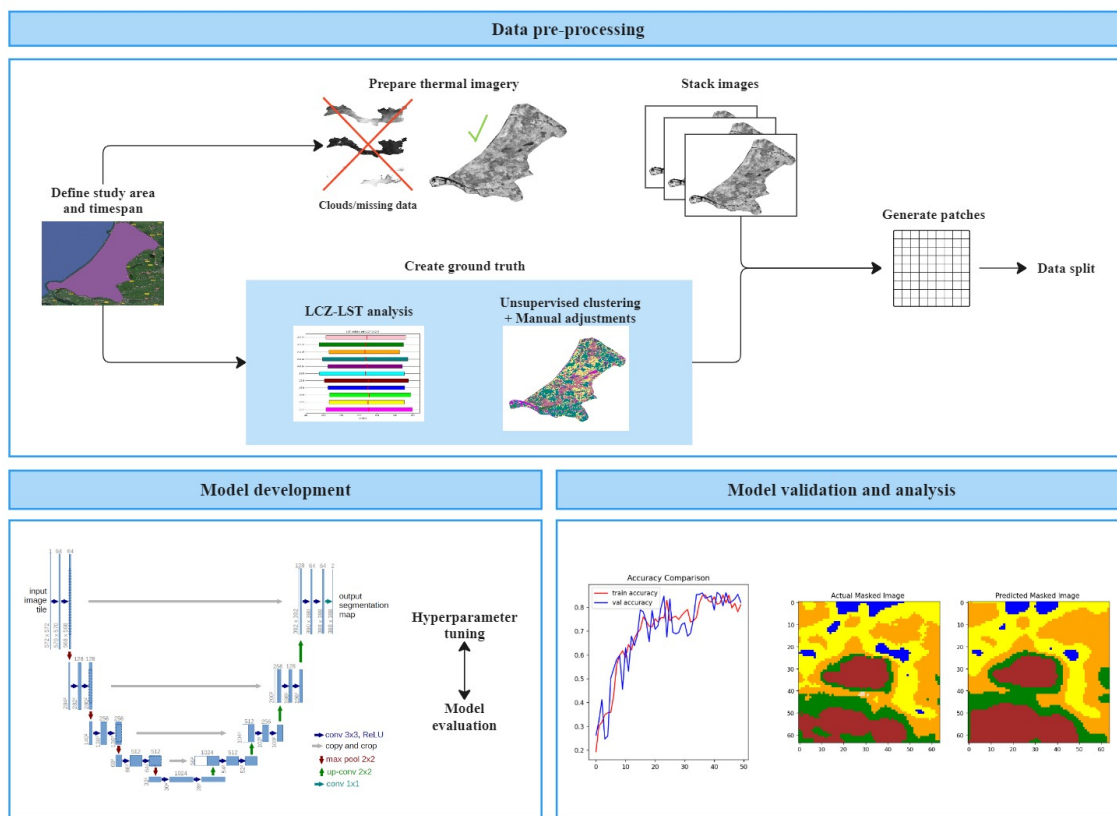


Figure 3.1.: Workflow

3. Methodology

3.1. Data pre-processing

This section will explain the data pre-processing step in the workflow. After performing these steps, the data is collected and prepared in such a way that it is ready for model utilization.

3.1.1. Study area and time span selection

In the initial phase, the study area and time span of the data area selected. They are selected by experimenting, having the following requirements in mind. Certain criteria must be met to ensure the suitability. The chosen area should have enough spatial extent to accommodate a substantial number of patches for the CNN model. However, it should not be excessively large to ensure frequent coverage within the satellite sensor's field of view, facilitating comparable LST measurements across the entire area. Moreover, the study area should exhibit diversity in terms of LCZs to enable the differentiation of various LCZs. The selected study area is shown in Figure 3.2. It ranges from Amsterdam to Rotterdam and therefore yields several large urban areas with varying distance from the coastline.

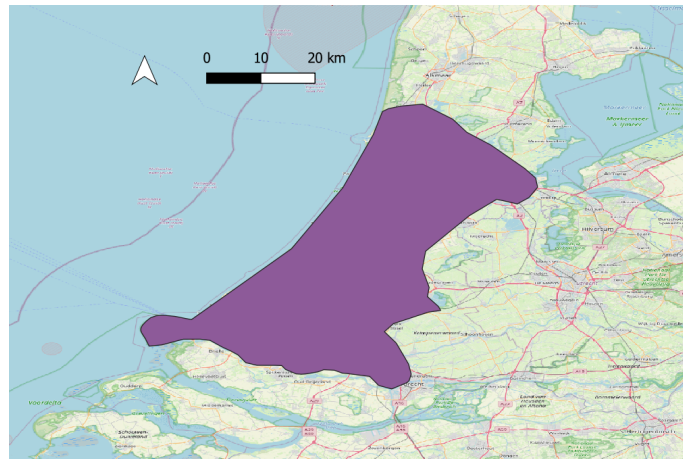


Figure 3.2.: Study area

Regarding the temporal aspect, there must be adequate availability of data during the selected time span. It is also beneficial to have images captured at different moments in time, to have visible temperature variations. For study area and time span both, computational complexity should be considered to prevent excessive resource consumption.

3.1.2. Data collection

For the selected study area and time span, thermal imagery will be retrieved. This thesis makes use of ECOSTRESS thermal imagery, which is open-source and has a frequent coverage. ECOSTRESS thermal imagery was introduced in Section 2.2. The imagery is accessed using the online open source tool: Application for Extracting and Exploring Analysis Ready Samples (AppEARS), that was developed by NASA. The data is downloaded by selected study

area and time span and manually processed. The following files are excluded by manual inspection from the final dataset:

- Thermal images that do not cover the entire study area. For every pixel the *LST* value at every measuring time is desired, to train the model with the same amount of input parameters per pixel.
- Thermal images with clouds. The thermal sensor cannot see through clouds. The study area is a coastal area where clouds are not uncommon.
- Thermal images with missing data. The data of *ECOSTRESS* is transferred in data bundles. Occasionally, a single data bundle is corrupted as it is transferred from the instrument to the ground data system. This can result in small patches of missing data within a file.
- Images with other artefacts. For example, occasionally the *ISS* must adjust the position of some of its solar panel arrays, these may pass into the *ECOSTRESS* field of view. This results in obstructions in the resulting images.

3.1.3. Stacking images

To put combined thermal datasets into the U-Net model, the rasters must be aligned and stacked into one raster with different time step values given as different band values. Every pixel has a vector of *LST* values from each of the input thermal images. Different stacks for different experiments are created.

3.1.4. Data normalization

After stacking the rasters, normalization is applied to each band independently in order to increase the speed of model convergence. The normalization was performed using linear scaling to re-map input values to a range between 0 and 1. The scaling function used is:

$$val_{out} = (val_{in} - c) \left(\frac{b - a}{d - c} \right) + a \quad (3.1)$$

Where val_{in} represents the original pixel index and val_{out} the normalized pixel value. a and b are the lower and upper values of the desired range, which are set to 0 and 1. c and d lower and upper values of the original range, which are set to the minimum and maximum of each band [Kadunc, 2022].

3.2. Training data labeling

Training data labeling is a critical component of the network development, because it ensures the quality and reliability of the model [Fredriksson et al., 2020]. In this thesis, the training data labeling is significantly more challenging, because there is no ground truth available. Ground truth refers to the accurate and reliable labeling or annotation of data, which serves

3. Methodology

as the reference against which the performance of the model is evaluated. There are several *LCZ* classifications available of the area of interest, but none of them based on thermal imagery.

Two different approaches to prepare the training data labeling are implemented:

- Analysis of *LST* time series data per *LCZ*
- From scratch with unsupervised clustering

The two approaches will be further explained in the following subsections.

3.2.1. *LCZ-LST* analysis

To analyze the thermal behaviour over time for different *LCZs*, a *LCZ* map of Europe created by Demuzere et al. [2019] will be used. 2 polygons per available class, of 140 m by 140 m (resembles 4 pixels for the *LST* data), will be drawn in each of the available *LCZs* in or close to the city of Rotterdam. The drawn polygons are shown in white in Figure 3.3. The available *LST* values inside the polygons over a time span of one year will be plotted and analyzed. The results will gain insight into the behavior of *LST* in different *LCZs* and help with the data labeling. It is expected that different *LCZs* show similar behavior, or that the same *LCZ* shows different behavior within the *LCZ*, because of the complex *LCZ-LST* relations [Lotfian et al., 2019]. If this hypothesis holds true, this means that not exactly the same classes as the map by Demuzere et al. [2019] will be used for the data labeling.

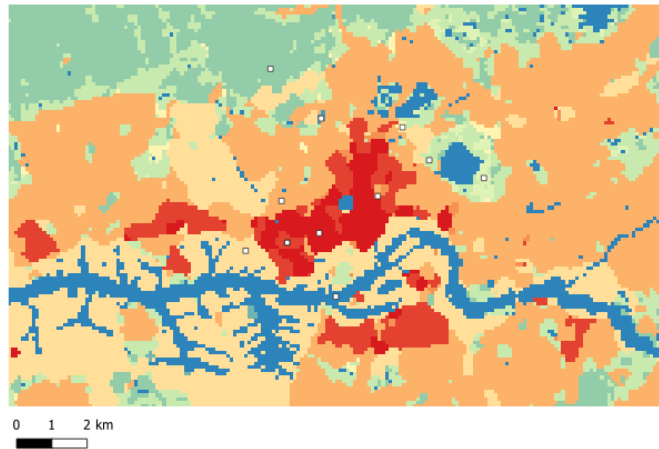


Figure 3.3.: Drawn polygons on *LCZ* map by Demuzere et al. [2019]

3.2.2. Unsupervised clustering methods

If the *LCZ-LST* analysis does not provide enough grip to distinguish every *LCZ* clearly from the other *LCZs* and to label the training data adequately, another method will be used. This option is feasible because *LCZ-LST* relations are complex and have an interconnected nature.

New LCZs, based on the thermal behaviour can be created from scratch, using unsupervised clustering, by grouping similar thermal timeseries together and discover underlying patterns.

The K-means clustering algorithm is applied to the pre-processed LST time series data. The choice of the number of clusters is first based on existing LCZ classifications, after which it is experimentally adjusted. Following the application of K-means, ISODATA is used to further refine the clustering results. As explained in Section 3.2.2, ISODATA adapts the number of clusters based on the data distribution, thereby enhancing the flexibility and robustness of the clustering process.

The clusters generated by K-means and refined by ISODATA are then subjected to manual inspection and labeling. The goal is to assign meaningful labels to each cluster, effectively categorizing the LST time series data based on their temperature patterns. Not a lot of interventions are done, because this clustering result is supposed to be a result of the LST time series and not one of existing LCZ knowledge.

3.3. Data split

The dataset was divided into training, test, and validation sets with a 70/15/15 split. This higher percentage for test and validation data, compared to the more common 10% in literature, was chosen due to the expected large number of classes. Ensuring that all classes are present in the test dataset makes it representative of the study area. A high-quality test dataset is crucial for evaluating the effectiveness of the classification model, as it provides a reliable basis for comparing predictions [Bai et al., 2021]. To ensure independent testing, 15% of the training dataset was reserved for evaluating the model on previously unseen images. The data split is done performed to prevent a biased distribution of data.

3.4. U-Net architecture

The U-net architecture employed in this research is adapted from the code described by Bhatia [2021] based on the original U-net architecture by Ronneberger et al. [2015] introduced in Section 2.4.5. The model is composed of five encoder blocks and five decoder blocks using ReLU as its activation function. For training, the model makes use of the Adam optimizer. The U-Net trains the detection of the classes present in the ground truth data in the training data set for a fixed number of epochs. The validation batches provide unbiased insights into training progress. Lastly, the model is trained, meaning that the weights are adjusted in such a way that the model should be able to classify LCZs in the testing data set. First, the model outputs a probability per class per pixel of the test data set that the pixel belongs to that class. The pixel will be assigned to the class that outputs the largest probability. The final classification outputted by the model is a multi-class LCZ prediction.

The determination of the hyperparameters patch size, loss function and learning rate will be performed by experimentation, to obtain the best result with the optimal hyperparameter combination.

3.5. Evaluation

The quality of the results can be evaluated in different ways. In order to evaluate the quality of the segmentation outputs, thermal images from the test datasets are provided to the U-net and the predicted masks are compared to the ground truth labels of these images. The following metrics are observed to evaluate the effectiveness of the model.

Accuracy, often called overall accuracy (OA), is a percentage that represents how many pixels are correctly classified from the total amount of pixels. The percentage is calculated by dividing the true positives (TP) (correctly classified pixels) by the total amount of pixels. Its equation is given by Equation 3.2. This metric provides an adequate indication of the overall performance of the model [Pedrayes et al., 2021].

$$OA = \frac{TP}{\text{Total of pixels}} \quad (3.2)$$

However, this thesis deals with a multi-class model. And the OA metric can be misleading if the classes are not balanced. For example, if one class represents the majority of pixels, it is also the main contributor to the OA. The other way around, if pixels from classes that are underrepresented are all classified wrongly, this might not show clearly in the OA percentage. Therefore, it is also desirable to consider the classes separately. The precision and recall are both calculated per class. Recall is defined by the percentage of correctly predicted pixels of a given class. It is calculated by dividing the TP by the sum of TP and false negatives (FN), given by Equation 3.3 [Pedrayes et al., 2021].

$$R = \frac{TP}{TP + FN} \quad (3.3)$$

Precision is defined by a percentage that represents how many classification predictions are correct from the total number of predictions for that class. It is calculated by dividing the TP by the sum of TP and false positives (FP), given by Equation 3.4 [Pedrayes et al., 2021].

$$P = \frac{TP}{TP + FP} \quad (3.4)$$

Then, with the precision and recall metrics, the F1-score can be computed per class. The F1-score is a metric that combines both recall and precision as a single value. It is computed as shown in Equation 3.5 [Pedrayes et al., 2021].

$$F1 = \frac{2TP}{2TP + FP + FN} = \frac{2 * R * P}{R + P} \quad (3.5)$$

Additionally, the macro F1-score metric is also computed. The F1-score metric was developed for single-label information retrieval, but there are variants of the F1-score for the multi-class models. Macro F1 calculates the F1 score for each class independently and then

averages these scores, giving equal weight to each class regardless of its size [Zhang et al., 2015].

$$\text{Macro F1} = \frac{1}{N} \sum_{i=1}^N \text{F1}_i \quad (3.6)$$

Another variant is the micro F1 score, which aggregates the individual contributions of all classes before calculating precision, recall, and F1 score. This means that it gives equal weight to each instance, thus being more influenced by the performance on larger classes. This number comes down to the same value as the computed OA [Zhang et al., 2015].

4. Implementation

4.1. Training data preparation

4.1.1. LCZ-LST analysis

As described in Section 3.2.1, a LCZ-LST analysis is performed to gain insight into the thermal behavior of LCZs in the study area of this thesis. Per available class in the classification map by [Demuzere et al., 2019], two polygons are drawn. The LST values of the pixels over a time span of one year of these polygons are downloaded and analyzed. The LST values are plotted in a box plot in Figure 4.1. In Figure 4.2, the same box plot is shown zoomed in. The LST mean, range, sample variance and standard deviation per LCZ can be found in Table 4.1.

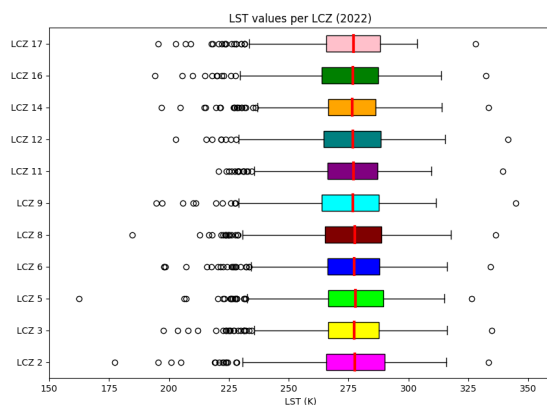


Figure 4.1.: box plot of LST per LCZ (2022)

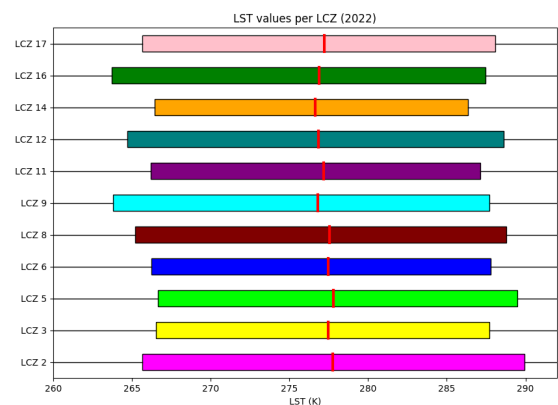


Figure 4.2.: box plot of LST per LCZ (2022) zoomed in

In the provided figures and the table, it is evident that there are differences in the thermal behavior between the classes. However, these differences do not significantly alter the overall range of LST values, indicating that the variation within each class may overlap, leading to similar thermal ranges across different classes. When analyzing the thermal behavior of LCZs, a consistent general trend is observed across the various classes. This means that although some LCZs may consistently exhibit higher or lower temperatures compared to others, the overall pattern of temperature change over time, such as diurnal cycles and seasonal variations, remains similar across different LCZs.

The Pearson correlation coefficient is also computed per polygon in relation to every other polygon. Even though some time series are generally warmer or cooler than time series from other LCZs, the Pearson correlation coefficient is sometimes higher between two pixels from

4. Implementation

Table 4.1.: LST mean, range, sample variance and standard deviation per LCZ

LCZ	Mean	Range	Sample variance	Standard deviation
2	275.30	156.13	573.75	23.95
3	275.10	137.18	527.51	22.97
5	275.09	164.14	527.87	22.98
6	274.33	136.34	531.02	23.04
8	275.31	151.77	540.79	23.25
9	274.03	150.19	495.32	22.26
A	274.17	118.81	368.16	19.19
B	275.10	138.52	488.91	22.11
D	274.06	136.84	466.58	21.60
F	274.62	216.46	573.82	23.95
G	273.58	132.66	448.62	21.18

different classes than between the two pixels from the same class. This leads to the conclusion that the training data is too intricate to be manually created due to the complexity of the time series. A ML model might be able to distinguish the different time series due to its ability to handle complex patterns and relationships in the data. However, for manual training data preparation, the time series are too complex, making it challenging to accurately categorize and analyze them without automated assistance. Therefore it is decided to first use unsupervised clustering and create new LCZs from scratch as training data for the model.

4.2. Training data set

As concluded in Section 4.1.1, first unsupervised clustering is performed on the stack of thermal raster files to create a labeled training data set. The methodology steps of the clustering are described in Section 3.2.2. The parameters for the K-means and ISODATA algorithms are experimentally determined. The ISODATA result that lead to the most satisfactory result had input values: 12 classes, 50 iterations, maximum standard deviation 0.0001 and minimum class size 10 pixels. The LST data of 3 years (2021, 2022 and 2023 respectively) is used for the clustering result. From this time span, 46 useful images were selected, excluding the files described in Section 3.1.2. The dates and times of the selected thermal images can be found in Appendix B.

The ISODATA result that was used to create the training data set is shown in Figure 4.3. Even though the input value for classes was 12, 10 distinguishable classes were created by the algorithm. The areas covered by the classes were analyzed with aerial imagery from Google Maps, and named by the observed dominant land cover. The class descriptions can be found in Table 4.2. The average LST value per thermal image per resulting cluster was plotted in Figure 4.4. These time series will be called thermal signatures.

Subsequently, the class representation is given in Table 4.3. It must be noted that the classes are not balanced and some classes contain significantly more pixels than other classes.

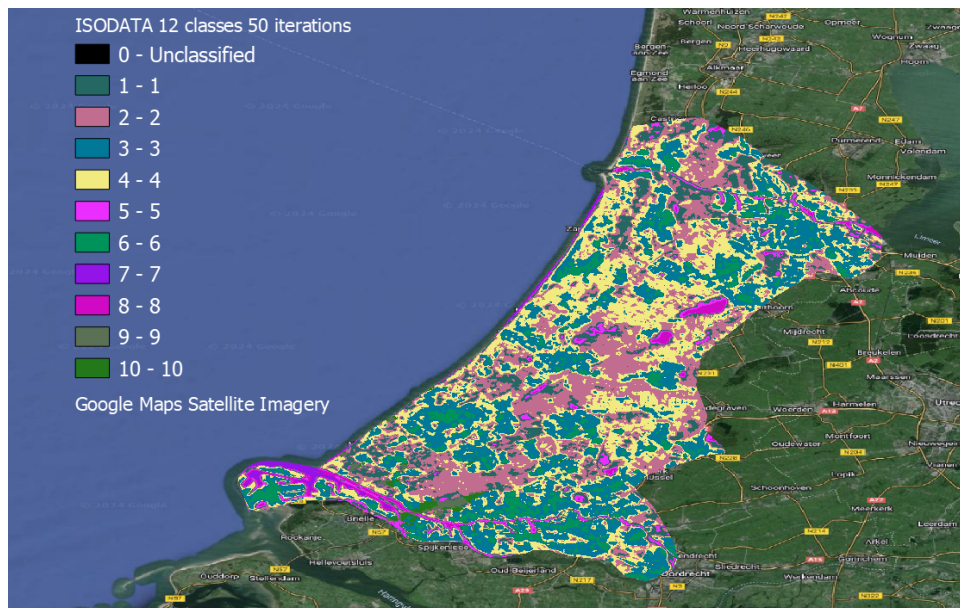


Figure 4.3.: ISODATA clustering results

Table 4.2.: Class descriptions based on aerial imagery

Class Number	Class Description
0	Unclassified
1	Dense forest/meadows, often next to water
2	Less dense forest/meadows
3	Residential area
4	Residential area with a lot of green/meadows
5	Shallow water
6	City centre/industrial area
7	Deepest water/sea water
8	Deep water
9	A few greenhouses, does not appear often
10	A few greenhouses, does not appear often

4. Implementation

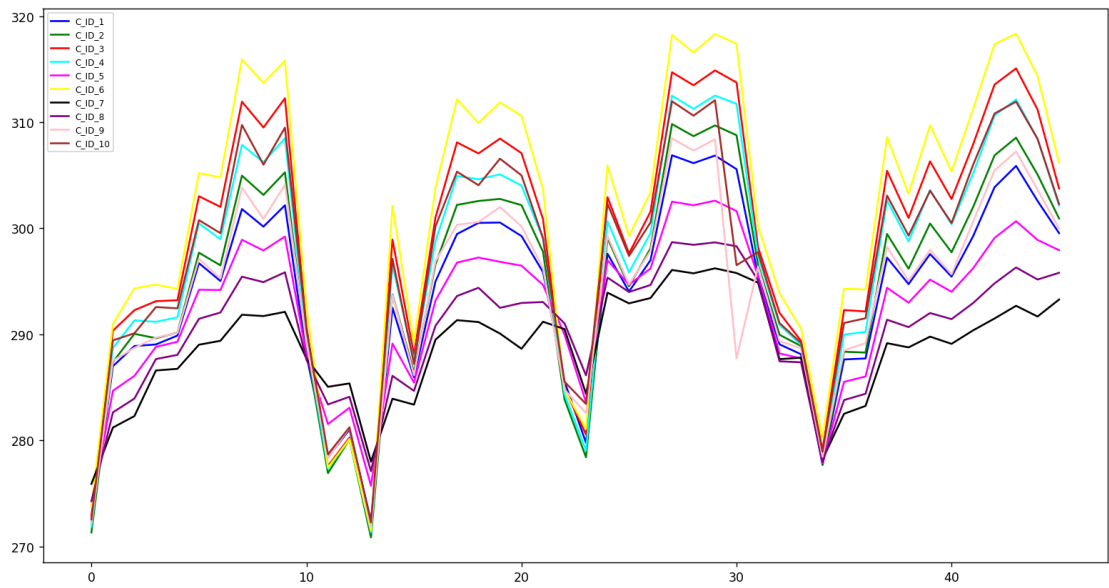


Figure 4.4.: Thermal signatures of resulting clusters from clustering algorithm

Class	Number of pixels	Percentage of total
1	52846	9.086
2	146680	25.22
3	154387	26.54
4	155032	26.65
5	13010	2.237
6	45612	7.842
7	845	0.145
8	5804	0.998
9	3339	0.574
10	4077	0.701

Table 4.3.: Class representation

4.3. Hyperparameter definition

A selection of hyperparameters has been tested to retrieve the best result for the model. The learning rate, patch size and loss function specifically. The values of hyperparameters are typically determined through experimentation and model performance evaluation [Durrani et al., 2023]. Therefore, different values for the learning rate and patch size are tested out, as well as different loss functions. One adjustment is made, while all the other parameters stay the same. The results are then evaluated.

4.3.1. Learning rate

The model's performance in terms of loss and accuracy are compared with attention on the learning rate to tune the learning process of the Adam optimizer. Learning rates of 0.1, 0.01, 0.001, and 0.0001 are implemented in the following figures. The loss function used in all runs is `SparseCategoricalCrossentropy()` and the patch size 64.

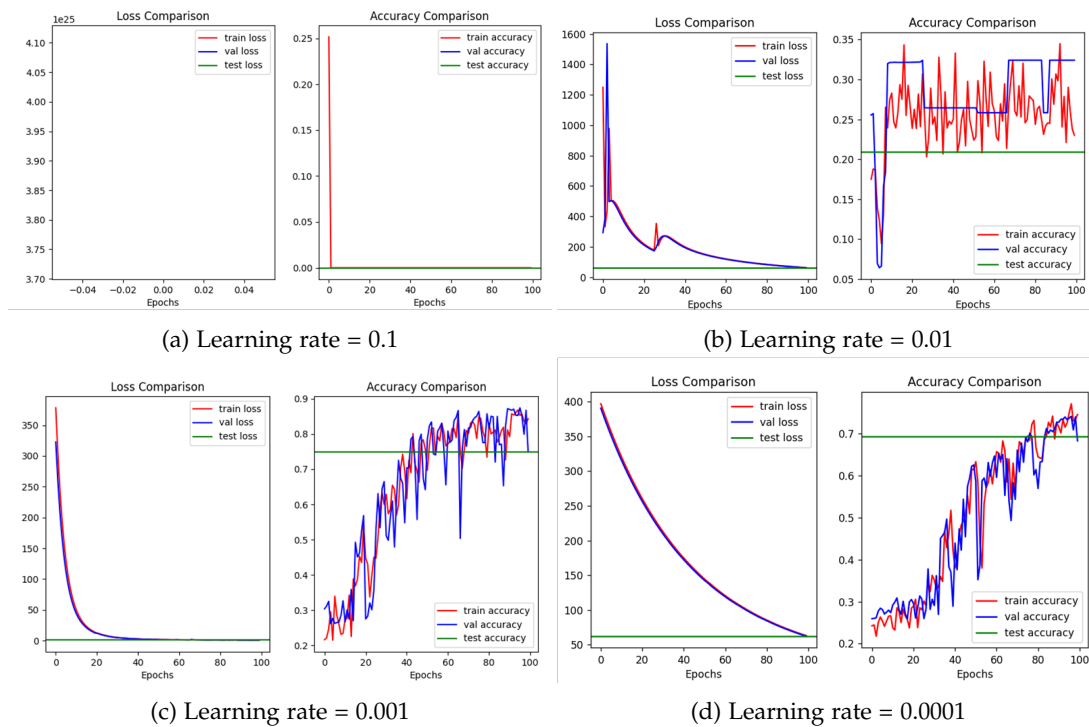


Figure 4.5.: Model's performance according to loss and accuracy with different learning rates

After conducting the series of experiments with learning rates of 0.1, 0.01, 0.001, and 0.0001, it was determined that a learning rate of 0.001 yields the best model performance. This learning rate was chosen because it was the only one where the loss function converged towards a constant value within 100 epochs. Additionally, the test accuracy achieved with a learning rate of 0.001 was the highest among all tested values (Table 4.4), indicating that the model not only learned effectively but also generalized well to new data. The other learning

4. Implementation

rates either caused significant fluctuations in the loss function or resulted in slow progress and sub-optimal accuracy. Therefore, a learning rate of 0.001 was selected to ensure efficient training and robust model performance.

Learning rate	Test accuracy
0.1	0.000
0.01	0.209
0.001	0.748
0.0001	0.694

Table 4.4.: Test accuracy values for different learning rates

4.3.2. Patch size

The model's performance in terms of loss and accuracy are compared with attention on the patch size. Patch sizes 64, 128 and 256 are implemented in the following figures. With a patch size of 64, the training, validation and test data is first pre-processed into patches of $64 \times 64 \times$ the number of bands. The loss function used in the runs is SparseCategorical-Crossentropy() and the learning rate the selected value of 0.001.

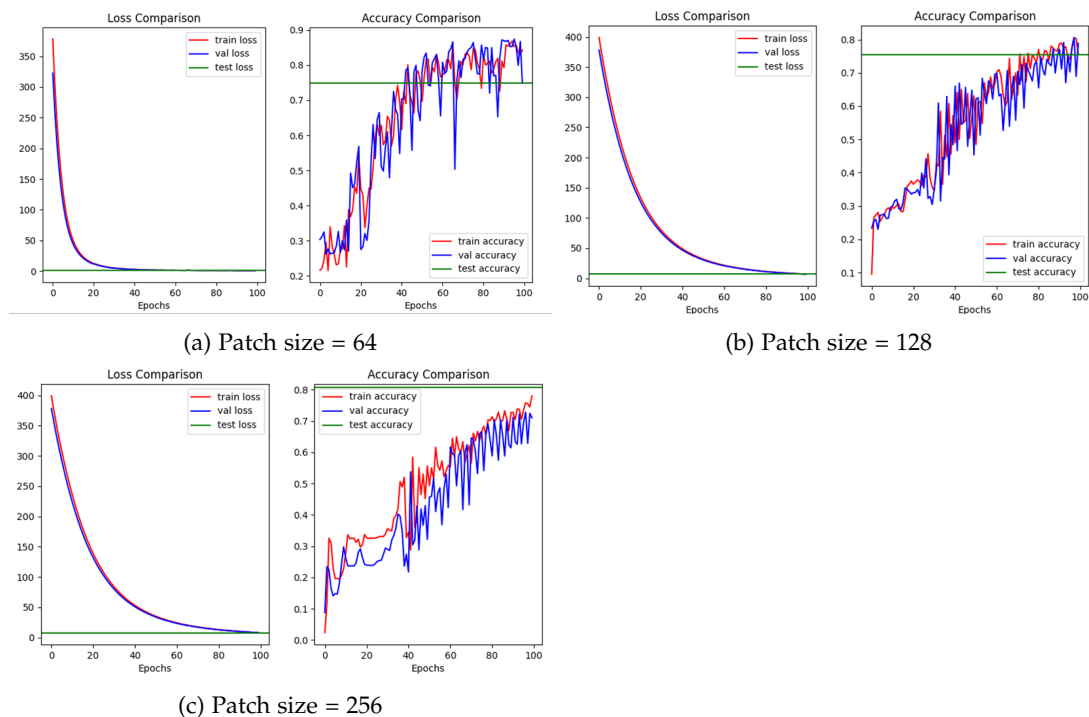


Figure 4.6.: Model's performance according to loss and accuracy with different patch sizes

After experimenting with patch sizes of 64, 128, and 256, it was determined that a patch size of 64 is the best choice for the model. Although a patch size of 256 yielded the highest

overall accuracy (Table 4.5), it learned much slower, with the loss function converging at a slower rate. Additionally, using such a large patch size resulted in fewer patches in the training data, as each 256x256 patch covers a significant area with a spatial resolution of 70-meter pixels. This limitation in the number of patches negatively impacted the training process. Therefore, a patch size of 64 was selected in the end, balancing learning speed and the amount of training data available, leading to a more efficient and robust model performance.

Patch size	Test accuracy
64	0.748
128	0.753
256	0.807

Table 4.5.: Test accuracy values for different patch sizes

4.3.3. Loss function

The model's performance in terms of loss and accuracy are also compared with attention the the loss function used. Two multi-class loss functions were tested out: `KLDivergence()` and `SparseCategoricalCrossentropy()` specifically. The patch size was set to 64 and the learning rate to 0.001. The results can be seen in Figure 4.7

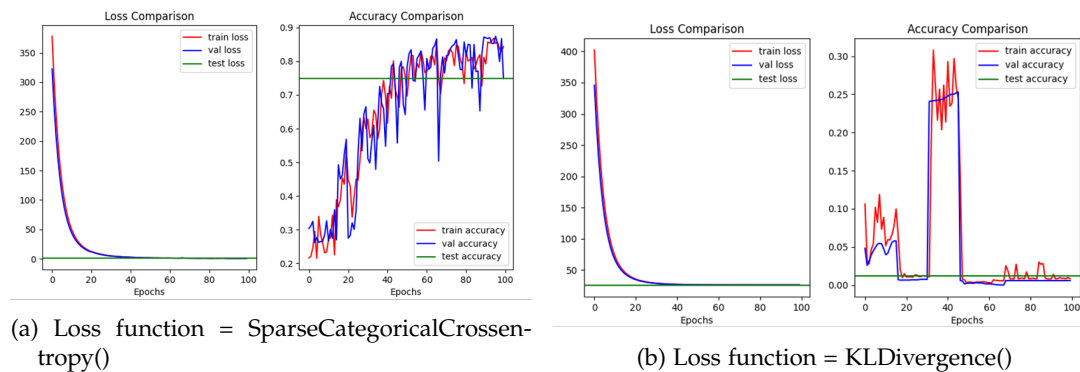


Figure 4.7.: Model's performance according to loss and accuracy with different loss functions

The model's performance when using the Kullback-Leibler Divergence loss function are not satisfactory, this can occur for several reasons. The gradients provided by `KLDivergence()` during backpropagation might be less informative or more unpredictable compared to those from `SparseCategoricalCrossentropy()`. This can make the optimization process less efficient, causing the model to struggle with learning the correct patterns from the data. Given these issues, `SparseCategoricalCrossentropy()` was selected for its more reliable and meaningful results. This loss function is well-suited for classification tasks, ensuring effective model performance and interpretability.

5. Results

5.1. Full dataset

In this section the performance of the model when being trained and tested with the full dataset of 46 thermal images will be presented and analyzed. The optimal hyperparameter combination that was selected in [Section 4.3](#) is used for this run: learning rate = 0.001, patch size = 64 and loss function = `SparseCategoricalCrossentropy()` respectively. 70% of the full dataset was used for training, 15% for validation and 15% for testing. The model performance can be found in [Figure 5.1](#).

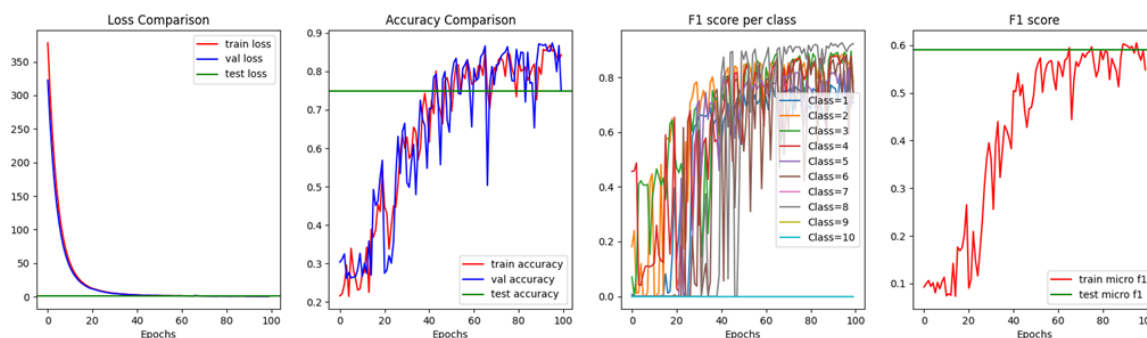


Figure 5.1.: Training and testing with full dataset results

It can be observed the loss functions converge smoothly, indicating effective learning and gradual optimization of the model parameters. The accuracy values and macro F1 score also converge towards a steady value, indicating that the model's performance stabilizes over time. The test accuracy value is 0.7484 specifically and the test macro F1 score is 0.5903. The macro F1 score value is significantly smaller than the accuracy value, because the macro F1 score gives each class the same weight. This means that the performance of some classes that do not appear often in the dataset is less favorable than the average performance of other classes. The "F1 score per class" graph shows the train F1 score per class per epoch. It is noticeable that some classes do not start performing better over the epochs. This can be explained by low representation of this class in the training data. The test F1 score per class values can be found in [Table 5.1](#)

It can be noticed that the majority of the classes provide a desirable test F1 score and class 7, 9 and 10 give a test F1 score of 0. These classes are defined as "Deepest water/sea water" and "A few greenhouses, does not appear often" for both class 9 and 10. This can be explained by low representation of these classes in the training and/or test data, or its complex nature. The class representation can be found in [Table 4.3](#). Class 7, 9 and 10 all have a representation smaller than 1% of the total number of pixels. This results in the model not distinguishing

5. Results

Class	Test F1 score per class
1	0.6976
2	0.8561
3	0.8877
4	0.8624
5	0.7649
6	0.8769
7	0.0000
8	0.9179
9	0.0000
10	0.0000

Table 5.1.: Test F1 score per class

these classes from other classes at all. The other classes yield more desirable test F1 scores, averaging between 0.7 and 0.9.

5.1.1. Probability distribution

In this subsection the probability distribution will be discussed. The model provides a probability value per class per pixel. The class with the maximum probability is the class that the pixel will be assigned to. This means the model can be more certain or less certain about a pixel's class. It was observed that the model performance differs per class. In [Figure 5.2](#) the distribution of the maximum probability value per class for the test data is shown. These distributions mirror the trends seen in the test F1 scores: When a class has a smaller F1 score, and is therefore more often misclassified than other classes, the model is also less certain of assigning the pixels to this class. This can be because class 7, 9 and 10 are underrepresented or they have complex behaviour to distinguish from other classes. Class 9 does not appear in the test data at all.

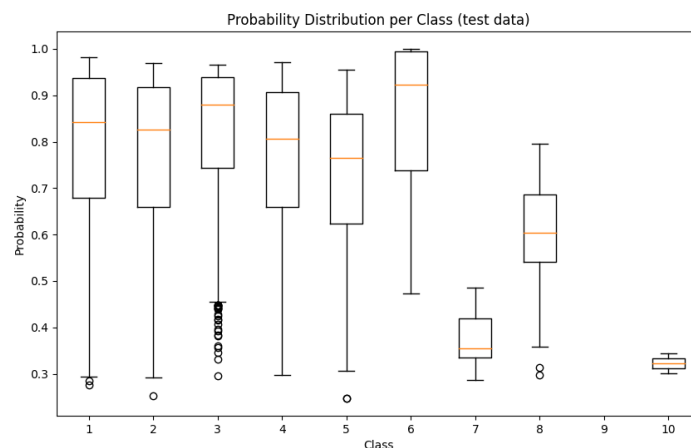


Figure 5.2.: Probability distribution per class for the test data

Figure 5.3 illustrates the spatial distribution of the average maximum probability across all classes within a test data patch. Edge effects can be noticed, indicating a degradation in the performance of the classification model towards the edges of the input data. This can be due to how the algorithm processes or analyzes the input data, leading to smaller classification probabilities for border pixels. The smaller certainty can be attributed to the reduced contextual information available at the patch edges, where pixels have fewer neighboring pixels for reference. It must be noted that a smaller maximum probability does not necessarily lead to a wrong pixel classification.

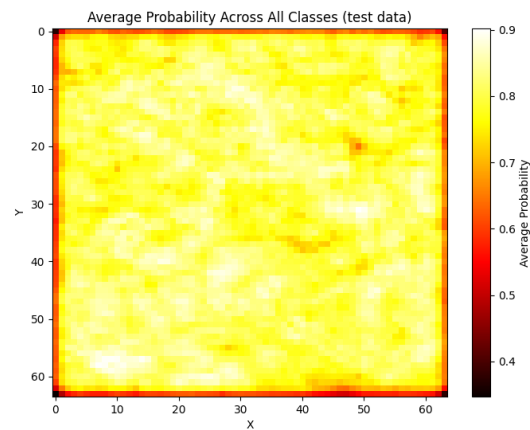


Figure 5.3.: Average maximum probability per test data pixel

5. Results

5.2. Seasonal influence

In this section the impact of using thermal imagery as training and testing dataset from different seasons is presented and analyzed. The initial training dataset was split in two parts: Spring/Summer and Autumn/Winter. These two subsets were used as training and testing input for two separate runs, using the optimal hyperparameters settings decided on in Section 4.3. When comparing the results of training and testing the model with these two datasets, the Spring/Summer dataset leads to better results, with larger values for the accuracy value and the F1 scores. The results of the two runs can be found in Figure 5.4 and Figure 5.5.

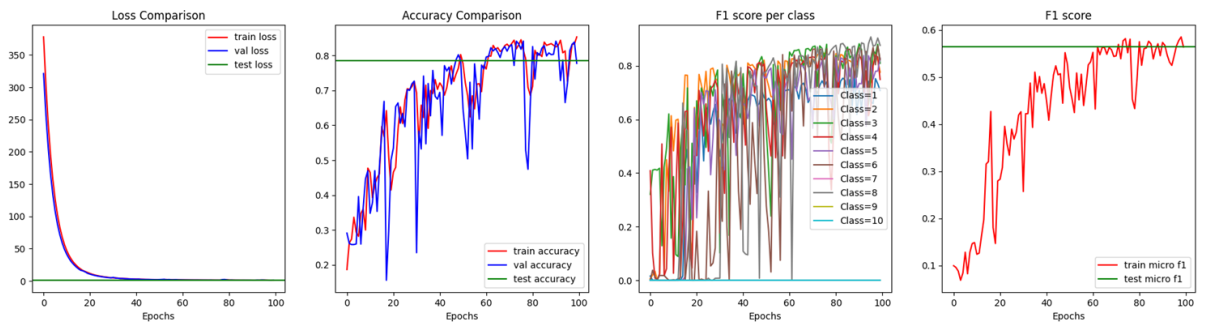


Figure 5.4.: Training and testing with data from Spring/Summer results

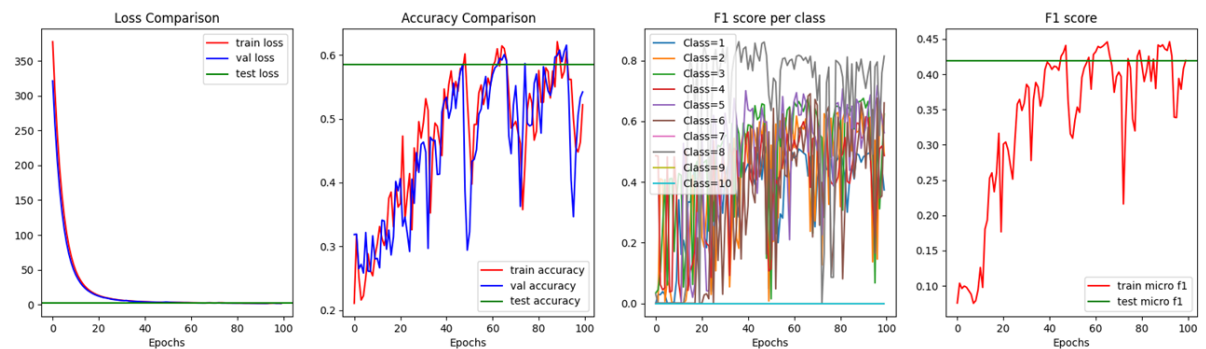


Figure 5.5.: Training and testing with data from Autumn/Winter results

It can be observed that in both runs, the loss functions converge smoothly, indicating effective learning and gradual optimization of the model parameters. The training and validation losses decrease and stabilize over time, suggesting that the model is not overfitting or underfitting significantly. When looking at the accuracy values, they also converge towards a steady value, indicating that the model's performance stabilizes over time. However, with the Autumn/Winter dataset, the accuracy values show greater variability across epochs, and more outliers can be observed. This suggests that the model may have more difficulty learning consistently from this dataset, and that the Autumn/Winter dataset might be less representative for the classification task, leading to inconsistent model performance. This is also emphasized by the smaller test accuracy value, specifically 0.5856 for Autumn/Winter

and 0.7850 for Spring/Summer. When looking at the test macro F1 score, the same relationship can be observed. The Autumn/Winter dataset results in a test macro F1 score of 0.4197, where the Spring/Summer dataset results in one of 0.5641. These values are smaller than the accuracy values, because each class has the same weight in this calculation. The unique F1 scores per class also provide an interesting insight, showing different performance per training and testing input dataset. Their values are shown in Table 5.2.

Class	Test F1 score per class	
	Spring/Summer	Autumn/Winter
1	0.7522	0.3374
2	0.8345	0.6245
3	0.8159	0.6596
4	0.7180	0.4392
5	0.7648	0.6781
6	0.8038	0.7308
7	0.0556	0.0000
8	0.7790	0.7011
9	0.0000	0.0000
10	0.0000	0.0000

Table 5.2.: Test F1 score per class for Spring/Summer and Autumn/Winter

It can be observed that almost all classes are classified better by the model that is trained and tested with the Spring/Summer dataset. This could be because there tend to be larger temperature variations between objects due to warmer ambient temperatures in Spring/Summer. This results in more distinct thermal signatures in the images, making it easier for the model to differentiate between different classes. This finding is also presented by Du et al. [2020], showing that LCZs are differentiated better in Summer than in other seasons regarding LST.

Some classes that are not classified better in Spring/Summer than in Autumn/Winter, are not classified (correctly) at all, with an F1 score of 0 or close to 0. This can be explained by the low representation of these classes in the training dataset. Class 7, 9 and 10 ("Deepest water/sea water", "A few greenhouses" and "A few greenhouses" respectively) have fewer pixels in the mask that is used for training. This low representation leads to poor model learning: With few examples, the model doesn't have enough data to learn the patterns and characteristics of that class effectively. The test F1 scores that are 0 can also be explained by the fact that the testing is done with only 15% of the initial dataset and sometimes this subset does not contain any pixels of these underrepresented classes. Overall, the model fails to classify certain classes effectively due to insufficient training data, but also the absence of these classes in the test subset.

Especially class 1 and 4 show a significant performance difference. Class 1 is described as "Dense forest/meadows, often next to water", and class 4 as "Residential area with a lot of green/meadows". It is possible that these classes are more difficult to distinguish when looking at only Autumn/Winter images, because it distinguishes itself from being relatively cooler compared to other classes in warmer periods. When training the model with only cooler images, this class might not be as distinguishable. In Figure 5.6 the actual mask of one patch with its classification result after training the model with Autumn/Winter images is shown. In the figure, class 1 is represented by blue and class 4 by orange. It can be

5. Results

seen that both these classes are generally misclassified as yellow, which is class 2 ("Less dense forest/meadows"). This implies that the percentage of tree coverage has less impact on the thermal signature in Autumn/Winter and is therefore less distinguishable. The classes that show the smallest performance difference are class 5 ("Shallow water"), class 6 ("City centre/industrial area") and class 8 ("Deep water"). This implies that these classes are comparably distinguishable in Autumn/Winter than in Spring/Summer.

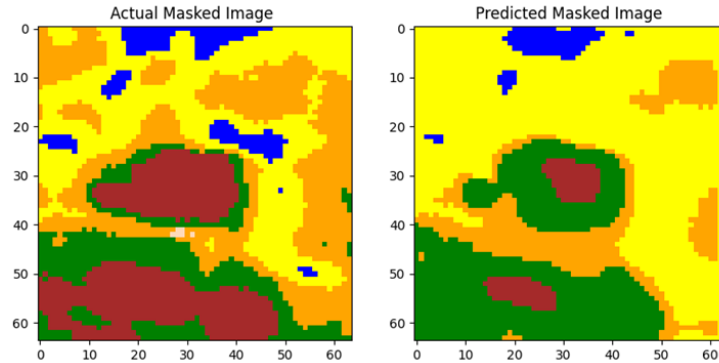


Figure 5.6.: Actual mask and classification result after training the model with Autumn/Winter images

It must be noted that the datasets used in this experiment are unbalanced. Specifically, the Autumn/Winter dataset comprises 7 images, whereas the Spring/Summer dataset contains 39 images. This imbalance can significantly influence the performance of the model. To mitigate this effect and ensure a fair comparison, additional experiments are conducted where two random subsets of 7 images were selected from the Spring/Summer dataset. These subsets were then used to match the number of images in the Autumn/Winter dataset, allowing to isolate and evaluate the impact of dataset size on model performance. The model performance of these runs is expressed in test accuracy in [Table 5.3](#).

Selection of Images	Number of Images	Test Accuracy
Spring/Summer	39	0.7850
Spring/Summer random selection 1	7	0.7585
Spring/Summer random selection 2	7	0.7831
Autumn/Winter	7	0.5856

Table 5.3.: Number of images and test accuracy for different selections

The values show that training and testing the model with a random selection of 7 Spring/Summer images does not change the larger test accuracy compared to training and testing the model with Autumn/Winter images. The performance of one of the two runs with a random selection of Spring/Summer images is also presented in [Figure 5.7](#) and [Table 5.4](#). Again, the same performance differences can be observed. This implies that the observed performance differences were a result of using images from different seasons rather than the number of images used.

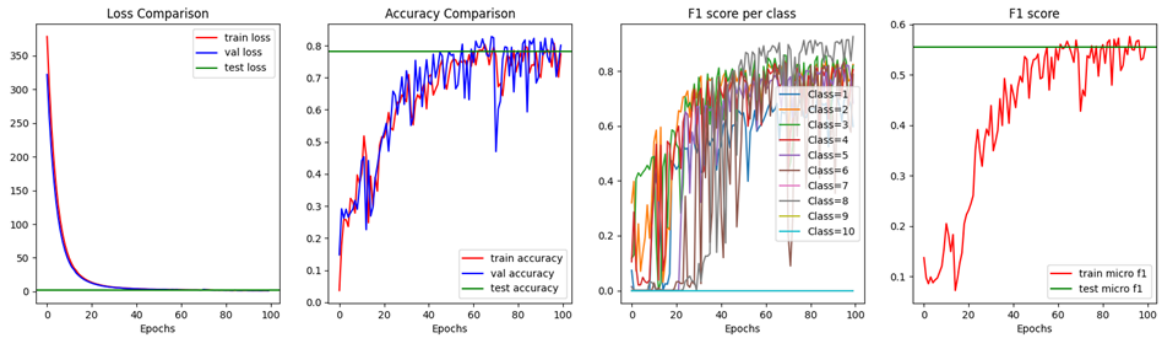


Figure 5.7.: Training and testing with a selection of 7 images from Spring/Summer results

Class	Test F1 score per class	
	Spring/Summer	Autumn/Winter
1	0.6230	0.3374
2	0.8253	0.6245
3	0.8307	0.6596
4	0.7581	0.4392
5	0.6948	0.6781
6	0.8052	0.7308
7	0.0000	0.0000
8	0.7714	0.7011
9	0.0000	0.0000
10	0.0000	0.0000

Table 5.4.: Test F1 score per class for Spring/Summer and Autumn/Winter, using 7 images

5.3. Daytime vs. nighttime

In this section the impact of using thermal imagery as training and testing dataset from different times (daytime vs. nighttime) is presented and analyzed. The initial training dataset was again split into two subsets, one consisting of thermal images taken at nighttime and one consisting of thermal images taken at daytime. The subsets were used as training and testing input for two separate runs, using the optimal hyperparameters settings decided on in Section 4.3. When comparing the results of the two runs, the daytime subset results in better model performance, with larger values for the accuracy and F1 scores. The results are shown in Figure 5.8 and Figure 5.9.

For these runs, it can also be observed that the loss functions converge smoothly, indicating effective learning and gradual optimization of the model parameters. The training and validation losses decrease and stabilize over time, suggesting that the model is not overfitting or underfitting significantly. When looking at the accuracy values, they also converge towards a steady value, indicating that the model's performance stabilizes over time. The performance of the nighttime run converges to a significantly smaller value than the daytime run, specifically 0.4764 and 0.8001. When looking at the test macro F1 score, the same correlation can be observed. The nighttime dataset results in a test macro F1 score of 0.2109 and the

5. Results

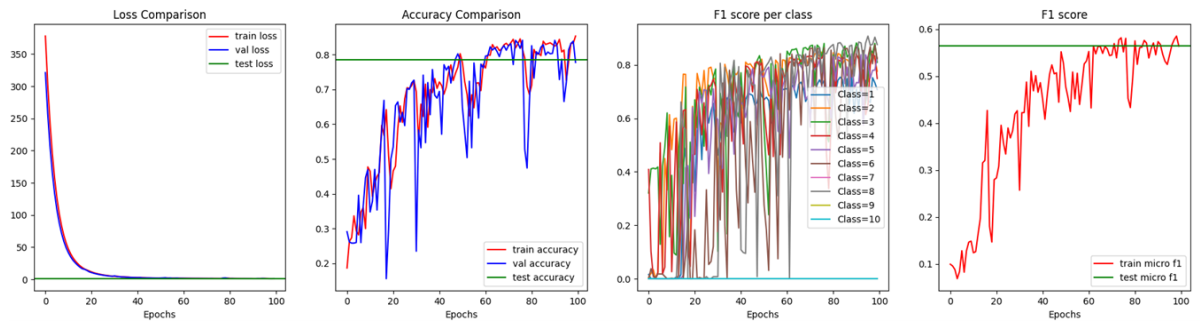


Figure 5.8.: Training and testing with data from daytime results

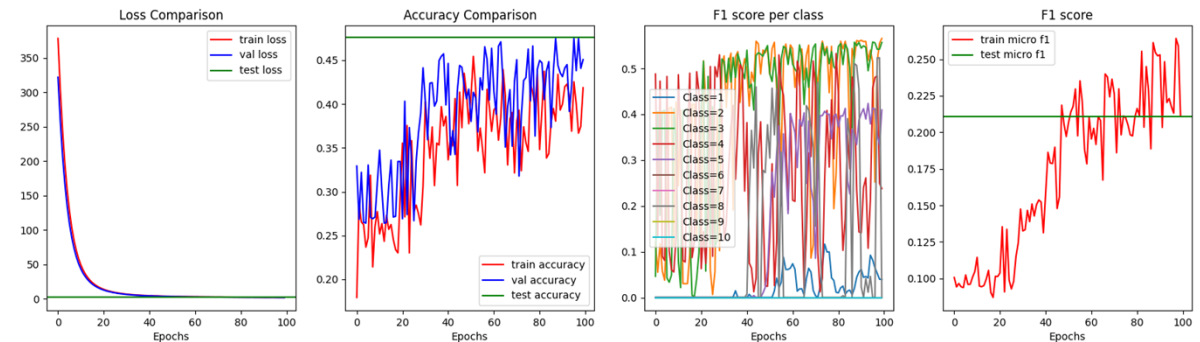


Figure 5.9.: Training and testing with data from nighttime results

daytime dataset in 0.5648. These values are smaller than the accuracy values, because each class has the same weight in this calculation. It is also important to look at the unique F1 scores per class, which show different performance per training and testing input dataset. Their values are shown in [Table 5.5](#).

It can be observed that almost all classes are classified better by the model that is trained and tested with the Spring/Summer dataset. The model performs generally worse for every class when using only night images. Half of the classes are not distinguished at all in the test images. The most significant performance differences between training/testing with images from daytime and training/testing with images from nighttime can be observed for the classes 1 ("Dense forest/meadows, often next to water"), 4 ("Residential area with a lot of green/meadows"), 6 ("City centre/industrial area") and 8 ("Deep water"). These classes were classified satisfactory with images from daytime, and with images from nighttime the test F1 scores are zero or close to zero. It is remarkable that these classes contain the generally cool (1, 4, 8) but also a generally warmer class (6). To get an idea to which other classes these classes are misclassified, the same patch and its predicted masked images is shown in [Figure 5.10](#). It can be seen that the red class ("City centre/industrial area") is misclassified as the green class (3, "Residential area") and a small part of the purple class (5, "Shallow water"). The yellow (2, "Less dense forest/meadows") and blue (1, "Dense forest/meadows, often next to water") is misclassified as orange (4, "Residential area with a lot of green/meadows"). A general trend that can be concluded, is that the classes that show more "extreme" behavior (warmer or cooler than other classes), are misclassified as

Class	Test F1 score per class	
	Daytime	Nighttime
1	0.7594	0.0513
2	0.8505	0.5874
3	0.8296	0.6596
4	0.7842	0.1409
5	0.7143	0.5035
6	0.7273	0.0000
7	0.0510	0.0000
8	0.7690	0.0000
9	0.0000	0.0000
10	0.0000	0.0000

Table 5.5.: Test F1 score per class for daytime and nighttime

classes with less fluctuations and more average values. This might be because at night the LST values of classes are more similar to each other.

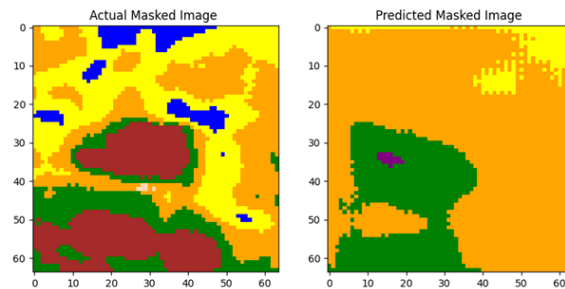


Figure 5.10.: Actual mask and classification result after training the model with nighttime images

It must be noted that the datasets used for this experiment are again unbalanced. For this experiment, there are only 5 images taken in nighttime, whereas the daytime dataset contains 41 images. To mitigate this effect and ensure a fair comparison, additional experiments are conducted where two random subsets of 5 images were selected from the daytime dataset. These subsets were then used to match the number of images in the nighttime dataset, allowing to isolate and evaluate the impact of dataset size on model performance. The model performance of these runs is expressed in test accuracy in Table 5.6.

A slight performance difference between the different random selections of 5 images and the dataset of 41 images can be noted, but a significant performance difference when only using nighttime images. This implies that the observed performance differences were a result of using images from different times rather than the number of images used.

5. Results

Selection of Images	Number of Images	Test Accuracy
Daytime	41	0.8001
Daytime random selection 1	5	0.7998
Daytime random selection 2	5	0.7657
Nighttime	5	0.4764

Table 5.6.: Number of images and test accuracy for different selections

5.4. Extreme analysis

In this section multiple subsets of the dataset were used as training and testing as an experiment. In the previous sections (Section 5.2 and Section 5.3) it was concluded that training and testing the model with images from days and on average warmer seasons lead to better results. When looking at the thermal signatures it can also be noticed that the thermal images with the highest *LST* values also show the biggest differences between thermal behaviour per class. Therefore the model was tested with only these images. The thermal signatures are shown in Figure 5.11. Four peaks in the average *LST* per image can be observed, they are shown in blue. It was also concluded from the previous experiments that using more images does not always lead to a better performance of the model. Therefore the model is trained and tested with different numbers of images from the observed peaks. With only one image (the image with the maximum *LST* values), one image per peak (four images with the highest *LST* values for that peak) and all images of the peaks (14 images) specifically.

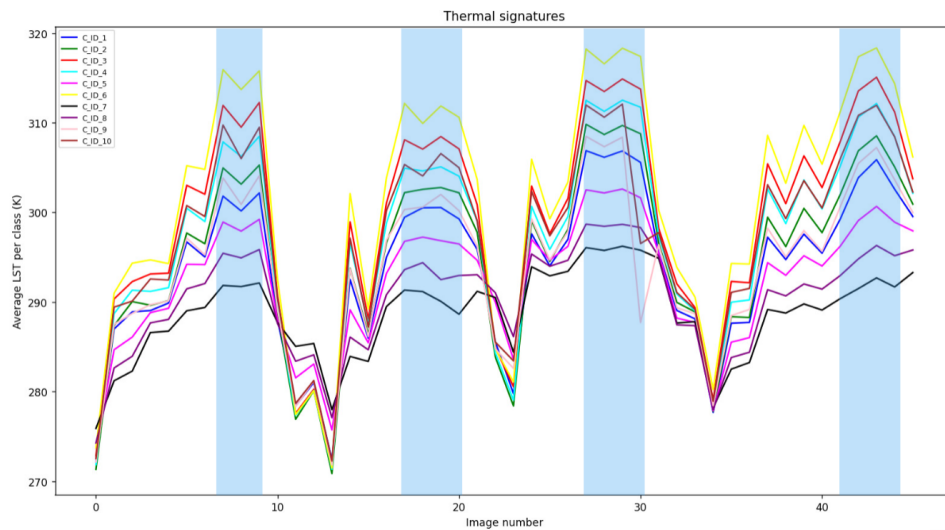


Figure 5.11.: Thermal signatures peaks

It can be noted that the test *OA* is comparable for the first two experiments and very high for the last experiment. In the first experiment, using a single image with high *LST* values, the test accuracy achieved is 0.260. This suggests that training and testing on a single image, while providing some discriminatory power, lacks generalizability due to limited data diversity. The second experiment includes four images with high *LST* values, the accuracy

Selection of images	number of images	Test accuracy
Maximum	1	0.260
Maximum per peak	4	0.285
All peaks	14	0.834

Table 5.7.: Test accuracy values for different image selections

improves slightly to 0.285. This increase is not significant but might indicate that incorporating more samples helps capture broader patterns. The dataset still falls short of robust performance. The third experiment, using 14 images with high *LST* values and emphasizing greater variability (not only the maximum values), significantly boosts accuracy to 0.83. This substantial improvement underscores the importance of dataset diversity in enhancing model generalization. By exposing the model to a wider range of *LST* variations across multiple images, it can learn more comprehensive features and achieve higher accuracy in classification tasks.

Remarkably, the accuracy of 0.83 in the third experiment surpasses that achieved when training and testing on the full dataset in Section 5.1, indicating that focusing on images with high *LST* values and ensuring variability can yield superior performance compared to a more generalized approach across all dataset samples.

6. Discussion, conclusion and future work

This chapter discusses the results presented in [Chapter 5](#) and draws conclusions to the research questions of the thesis. The limitations encountered in this research are also summarized, and potential future research following on this thesis is presented.

6.1. Discussion

The [CNN](#) model with U-net architecture presented in [Chapter 3](#) has been developed and optimized. In [Chapter 5](#), the model is trained and tested with different datasets.

The results of this study indicate that spatio-temporal thermal imagery, combined with the U-net architecture, is effective in classifying urban [LCZs](#). The significant impact of temporal factors, such as diurnal and seasonal variations, underscores the importance of considering time-series data in [LCZ](#) classification. The observed higher performance with daytime and Spring/Summer imagery suggests that these conditions provide clearer, more distinguishable thermal signatures, facilitating better differentiation between [LCZs](#). The findings align with previous research indicating the utility of remote sensing data in urban climate studies. Unlike traditional [LCZ](#) classification methods that rely on multi-spectral data, this approach focuses on thermal behavior, offering a new perspective. The improved classification accuracy during daytime and warmer seasons supports studies by [Stewart and Oke \[2012\]](#), [Lotfian et al. \[2019\]](#) and [Zhao et al. \[2021\]](#), which emphasize the role of thermal properties in defining urban [LCZs](#). The ability to distinguish [LCZs](#) based on their thermal behavior provides urban planners with valuable insights into the thermal characteristics of different urban areas. This information can inform strategies to mitigate urban heat islands and enhance urban resilience to climate change.

6.2. Limitations

This section outlines the key limitations encountered in this study, which could impact the generalizability and robustness of the findings.

Temporal Resolution: The study relies on data from the [ECOSTRESS](#) sensor aboard the [ISS](#). Despite the [ISS](#) providing daily coverage, the temporal resolution is hindered by two main issues. First, the known operational challenges with [ECOSTRESS](#), as discussed in [Chapter 3](#), limit the availability and consistency of data. Second, thermal sensors are unable to see through cloud cover, further reducing the number of usable images. Consequently, this results in temporal gaps in the dataset, potentially affecting the model's ability to accurately capture and classify temporal variations in [LST](#).

Spatial Resolution: The thermal imagery used in this study has a spatial resolution of 70 meters by 70 meters. While this resolution is adequate for capturing broader thermal

6. Discussion, conclusion and future work

patterns, it falls short for detailed local site studies. Small urban features may not be distinguishable at this scale. This limitation could lead to inaccuracies in classifying densely built-up areas and small-scale urban phenomena, reducing the precision of the [LCZ](#) classification in those contexts.

Non-standardized Labeling: The [LCZ](#) labeling system developed in this research is created to the specific thermal characteristics observed in the study area and is not standardized. Unlike the widely recognized [LCZ](#) classification system proposed by [Stewart and Oke \[2012\]](#), the labels used here are based on clusters derived from the thermal behavior of the study area. This non-standardized approach makes it challenging to compare results with other studies or to generalize the findings to different geographic areas. The lack of standardization could also hinder the integration of this research into broader urban climate studies.

Training Data Bias: The training dataset used to develop the classification model may be biased towards Spring/Summer conditions. If the majority of the training images represent warmer months, the model might learn seasonal-specific patterns, leading to better performance on similar Spring/Summer images and poorer performance on Autumn/Winter images. This seasonal bias can result in overfitting, where the model performs well on the training data but fails to generalize across different seasonal conditions. Addressing this bias requires a more balanced training dataset that includes representative samples from all seasons.

Weather Conditions: The model's performance might be influenced by varying weather conditions, which were not explicitly accounted for in this study. Factors such as humidity, wind speed, and precipitation can affect thermal imagery and [LST](#) values. The absence of weather condition data in the analysis could limit the model's ability to accurately classify [LCZs](#) under diverse conditions. Future studies should consider incorporating weather data to enhance the robustness of the classification model.

In summary, while this study is a good starting point in utilizing spatio-temporal thermal imagery for [LCZ](#) classification, these limitations highlight areas for improvement. Addressing these issues in future research will enhance the reliability and applicability of the findings across different urban environments and climatic conditions.

6.3. Conclusion

This section addresses each of the research sub-questions and concludes by drawing conclusions to the main research question for this thesis.

- How can a representable training data set be collected?

In this research, the training data labeling was challenging, because there is no ground truth available. Existing [LCZ](#) classifications are based on multi-spectral data, multi-spectral data enriched with vector and raster data, or manual sampling, but not based on the thermal behavior. In this thesis it was concluded that the time series of [LST](#) behavior per [LCZ](#) in the study area was too complex for manual training data preparation, even though a [ML](#) model might be able to distinguish the different series. Therefore a training data set was collected by applying the [ISODATA](#) algorithm to a stack of thermal images at different times. The thermal images were first manually selected, excluding images with issues. The resulting clusters were slightly adjusted manually, analyzed and described based on aerial imagery from Google Maps. The resulting clusters are

not the same classes as in the LCZ classification system by Stewart and Oke [2012], but they are distinguishable based on their thermal behavior. The classes still contain different land cover types and their differences in thermal behavior makes them valuable categories for urban planning. The mask was split in training data, validation data and test data. All in all, a training data set was collected that was representable for the application of this thesis. When comparing or collaborating with other fields of study, more standardized labels may be desired.

- When it comes to the architecture of U-net, what values for the hyperparameters of the deep learning network lead to the best classification result?

Different hyperparameters were determined through experimentation and model performance evaluation in Section 4.3. The learning rate, patch size and loss function specifically. Different values for the learning rate and patch size were tested out, as well as different loss functions. One adjustment is made, while all the other parameters stay the same. The other adjustable parameters of the architecture of the U-net were adopted from the U-net described by Bhatia [2021], based on the original U-net architecture by Ronneberger et al. [2015].

By experimenting it was determined that a learning rate of 0.001 yields the best model performance. This learning rate was chosen because it was the only one where the loss function converged towards a constant value within 100 epochs. Additionally, the test accuracy achieved with a learning rate of 0.001 was the highest among all tested values, indicating that the model not only learned effectively but also generalized well to new data.

After experimenting with different patch sizes, it was determined that a patch size of 64 is the best choice for the model. Although a patch size of 256 yielded the highest OA, it learned much slower, with the loss function converging at a slower rate. Additionally, using such a large patch size resulted in fewer patches in the training data, as each 256x256 patch covers a significant area with a spatial resolution of 70-meter pixels. This limitation in the number of patches negatively impacted the training process. Therefore, a patch size of 64 eventually selected, balancing learning speed and the amount of training data available, leading to a more efficient and robust model performance.

For the loss function, SparseCategoricalCrossentropy() was selected for its more reliable and meaningful results. This loss function is well-suited for classification tasks, ensuring effective model performance and interpretability.

- What is the impact of temporal frequency (day-night, seasonal) on the classification performance?

In Chapter 5, multiple experiments were conducted by training and testing the model with subsets of the full dataset, specifically day-night subsets and seasonal subsets, and subsets with maximum LST values. The experiments revealed that temporal factors significantly influence the effectiveness of the thermal imagery-based classification model. When comparing daytime and nighttime thermal imagery, it was observed that daytime images resulted in higher model performance. The same was the case for Spring/Summer images compared to Autumn/Winter images. The daytime and Spring/Summer dataset led to larger values for both OA and F1 scores compared to the nighttime and Autumn/Winter dataset. This disparity is likely due to the higher

6. Discussion, conclusion and future work

thermal contrast and more distinct thermal signatures present in daytime and Spring/-Summer images, which facilitate better differentiation between different LCZs.

In summary, these experiments underscored that temporal factors, such as variations in LST values, profoundly influence the effectiveness of the thermal imagery-based classification model. Optimal model performance is achieved when datasets capture diverse seasonal and extreme conditions, enabling the model to learn comprehensive thermal signatures and improve classification outcomes across different LCZs. Even though the diversity leads to distinguishable LCZs, more images did not necessarily lead to the best result.

The answers to the sub-questions lead to the conclusion of the main research question: *To what extent is a CNN with U-net architecture using spatio-temporal thermal imagery suitable for the classification of urban LCZs?*

This thesis demonstrates that the developed CNN with U-net architecture, trained with spatio-temporal thermal imagery, is suitable for the classification of urban LCZs. The key factors contributing to this suitability include:

- An effective method for collecting a representable training dataset tailored to thermal behaviors in urban areas.
- Optimized hyperparameters that enhance model learning and generalization.
- A significant impact of temporal diversity on classification performance, emphasizing the importance of including varied temporal conditions in the dataset.

Overall, the approach proves to be robust and adaptable, offering valuable insights for urban planning and environmental monitoring. This thesis has explored the utilization of a new source for LCZ classification, providing a useful starting position for further enhancement of standardized LCZ classification.

6.4. Future work

In light of the findings from this research, several avenues for future work can be explored to enhance the accuracy and robustness of LCZ classification using thermal imagery, some already presented in Section 6.2.

Weather Conditions: One critical aspect to consider in future research is the incorporation of weather conditions. The LST trends observed in the thermal images are significantly influenced by weather patterns, such as cloud cover, precipitation, and wind speed. By integrating meteorological data into the analysis, it is possible to better understand and account for these influences, thereby improving the model's ability to accurately classify land cover types under varying weather conditions.

Integration with Other Geospatial Data: Another promising direction is the combination of thermal imagery with other geospatial data sources. Utilizing additional data such as multispectral imagery, topographic information, and land use/land cover maps can provide complementary information that enhances the classification process. This multi-source approach can refine LCZ classification by leveraging the strengths of different data types, resulting in a more comprehensive and accurate understanding of land cover characteristics.

Integration of other thermal imagery sources: The temporal resolution of thermal imagery is a crucial factor in capturing dynamic changes in land cover. Future work could explore the integration of [ECOSTRESS](#) thermal imagery with other sources of thermal data to increase the frequency of observations. Combining datasets from multiple sources can help fill temporal gaps and provide a more continuous and detailed temporal record of [LST](#) variations. The different spatial resolution might be a challenge to overcome.

A. Reproducibility self-assessment

A.1. Marks for each of the criteria

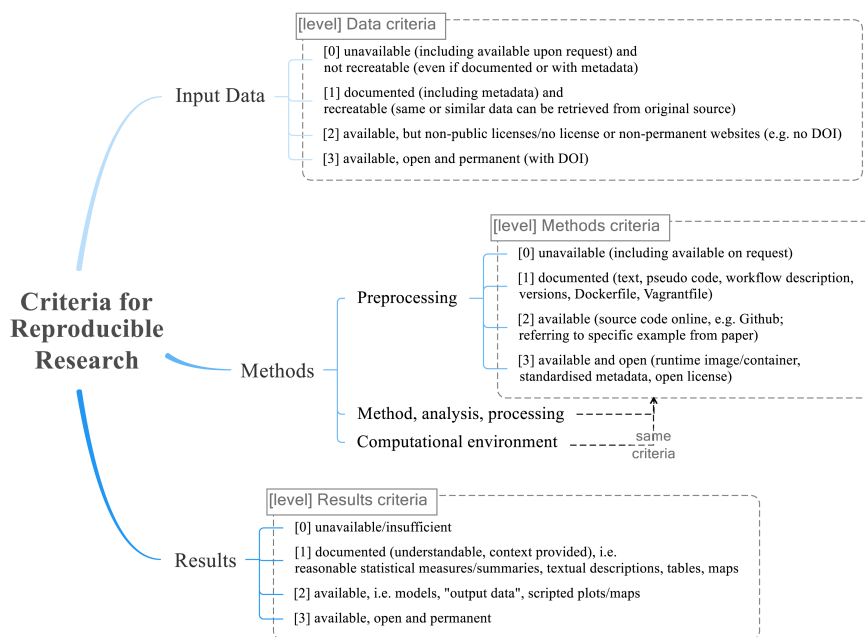


Figure A.1.: Reproducibility criteria

Category	Criteria	Grade
1. Input data	Satellite imagery	3
	Ground truth	0
2. Methods	Pre-processing	1
	Method, analysis, processing	1
	Computational environment	3
3. Results		1

Table A.1.: Evaluation of reproducibility criteria

A.2. Self-reflection

This chapter provides a self-reflection about the reproducibility of this thesis, which is divided into the criteria of input data, methods, and results. The assigned grades in [Table A.1](#) are clarified per category in this section.

The first criterion concerns the input data, consisting of the thermal imagery and the ground truth masks. The [ECOSTRESS](#) thermal imagery data is publicly available, and free of charge, via the [AppEARS](#) tool. In contrast, the ground truth datasets are manually generated and not publicly available.

The implemented pre-processing steps and analyses are reproducible by following the documentation provided by this thesis in [Chapter 3](#) and [Chapter 4](#). The computational environment exists of freely usable software, specifically QGIS and the programming language Python using PyCharm. All the packages and environments used where also open-source.

The results are documented and described in [Chapter 5](#) in the form of tables, plots and graphs. Textual descriptions and reasonable summaries are provided.

B. Dates and times thermal images

B. Dates and times thermal images

Table B.1.: Dates and times of selected thermal images

Day-Month	Year	Time
11-02	2021	11:39:05
30-03	2021	17:04:29
20-04	2021	08:34:25
29-05	2021	17:06:05
30-05	2021	17:55:16
01-06	2021	16:20:50
06-06	2021	15:36:58
10-06	2021	12:27:45
13-06	2021	11:42:23
14-06	2021	10:54:54
16-08	2021	13:40:36
24-10	2021	06:42:52
28-10	2021	06:47:33
26-02	2022	06:56:24
17-04	2022	11:02:15
23-04	2022	07:48:13
04-06	2022	15:00:50
15-06	2022	10:59:21
17-06	2022	09:23:11
17-06	2022	10:59:29
18-06	2022	10:11:35
22-06	2022	08:35:36
02-07	2022	03:47:31
04-07	2022	03:47:50
01-08	2022	15:41:46
01-08	2022	17:18:19
02-08	2022	16:30:08
12-08	2022	11:37:16
13-08	2022	10:49:07
13-08	2022	12:25:49
14-08	2022	11:37:20
22-08	2022	08:21:59
09-10	2022	12:08:16
18-10	2022	09:43:09
08-02	2023	13:32:40
11-04	2023	13:01:07
19-04	2023	09:48:25
03-06	2023	14:21:31
03-06	2023	15:57:53
06-06	2023	13:31:44
06-06	2023	15:08:12
08-06	2023	13:30:23
09-06	2023	12:41:34
11-06	2023	11:03:10
13-06	2023	11:02:05
25-06	2023	07:44:46

Bibliography

- Aggarwal, K., Mijwil, M. M., Sonia, Al-Mistarehi, A. H., Alomari, S., Gök, M., Alaabdin, A. M. Z., and Abdulrhman, S. H. (2022). Has the future started? the current growth of artificial intelligence, machine learning, and deep learning. *Iraqi Journal for Computer Science and Mathematics*, 3:115–123. doi:10.52866/ijcsm.2022.01.01.013.
- Aghabozorgi, S., Seyed Shirshorshidi, A., and Ying Wah, T. (2015). Time-series clustering – a decade review. *Information Systems*, 53:16–38. doi:10.1016/j.is.2015.04.007.
- Ajit, A., Acharya, K., and Samanta, A. (2020). A review of convolutional neural networks. *International Conference on Emerging Trends in Information Technology and Engineering, ic-ETITE 2020*. doi:10.1109/IC-ETITE47903.2020.049.
- Aslam, A. and Rana, I. A. (2022). The use of local climate zones in the urban environment: A systematic review of data sources, methods, and themes. *Urban Climate*, 42:101120. doi:10.1016/J.UCLIM.2022.101120.
- Bai, Y., Chen, M., Zhou, P., Zhao, T., Lee, J., Kakade, S., Wang, H., and Xiong, C. (2021). How important is the train-validation split in meta-learning? In Meila, M. and Zhang, T., editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 543–553. PMLR. URL <https://proceedings.mlr.press/v139/bai21a.html>.
- Ball, G. H. and Hall, D. J. (1965). Isodata, a novel method of data analysis and pattern classification. URL <https://api.semanticscholar.org/CorpusID:53887616>.
- Baskin, C., Liss, N., Zheltonozhskii, E., Bronshtein, A. M., and Mendelson, A. (2017). Streaming architecture for large-scale quantized neural networks on an fpga-based dataflow platform. doi:10.1109/IPDPSW.2018.00032.
- Bhatia, V. (2021). U-net implementation from scratch using tensorflow. URL <https://medium.com/geekculture/u-net-implementation-from-scratch-using-tensorflow-b4342266e406>.
- Bishop, C. M. (2006). Pattern recognition and machine learning. *Information Science and Statistics*, page 738. URL <https://www.springer.com/gp/book/9780387310732http://users.isr.ist.utl.pt/~wurmd/Livros/school/Bishop-PatternRecognitionAndMachineLearning-Springer2006.pdf>.
- Bishop, C. M. and Bishop, H. (2023). *Deep Learning: Foundations and Concepts*. Springer International Publishing. doi:10.1007/978-3-031-45468-4.
- Bracewell, R. (1999). *The Fourier Transform And Its Applications*. McGraw-Hill Higher Education, third edition.

Bibliography

- Bradski, G. and Kaehler, A. (2008). *Learning OpenCV: Computer vision with the OpenCV library*. O'Reilly Media, Inc. URL https://books.google.nl/books?hl=nl&lr=&id=seAgi0fu2EIC&oi=fnd&pg=PR3&ots=hWE48ncHN9&sig=Vjol5NejWfwxUmiahPOGBUZ1Rk4&redir_esc=y#v=onepage&q&f=false.
- Briottet, X., Chehata, N., Oltra-Carrio, R., Le Bris, A., and Weber, C. (2016). 1 - optical remote sensing in urban environments. In Baghdadi, N. and Zribi, M., editors, *Land Surface Remote Sensing in Urban and Coastal Areas*, pages 1–62. Elsevier. doi:10.1016/B978-1-78548-160-4.50001-7.
- Chen, Y. C., Chiu, H. W., Su, Y. F., Wu, Y. C., and Cheng, K. S. (2017). Does urbanization increase diurnal land surface temperature variation? evidence and implications. *Landscape and Urban Planning*, 157:247–258. doi:10.1016/J.LANDURBPLAN.2016.06.014.
- Choi, R. Y., Coyner, A. S., Kalpathy-Cramer, J., Chiang, M. F., and Campbell, J. P. (2020). Introduction to machine learning, neural networks, and deep learning. *Translational Vision Science Technology*, 9:14–14. doi:10.1167/TVST.9.2.14.
- Cilek, M. U. and Cilek, A. (2021). Analyses of land surface temperature (lst) variability among local climate zones (lczs) comparing landsat-8 and envi-met model data. *Sustainable Cities and Society*, 69:102877. doi:10.1016/J.SCS.2021.102877.
- Demuzere, M., Bechtel, B., Middel, A., and Mills, G. (2019). Mapping europe into local climate zones. *PLOS ONE*, 14:e0214474. doi:10.1371/JOURNAL.PONE.0214474.
- Du, P., Chen, J., Bai, X., and Han, W. (2020). Understanding the seasonal variations of land surface temperature in nanjing urban area based on local climate zone. *Urban Climate*, 33:100657. doi:10.1016/j.uclim.2020.100657.
- Durrani, A. R., Minallah, N., Aziz, N., Frnda, J., Khan, W., and Nedoma, J. (2023). Effect of hyper-parameters on the performance of convlstm based deep neural network in crop classification. *PLOS ONE*, 18:e0275653. doi:10.1371/JOURNAL.PONE.0275653.
- Fredriksson, T., Mattos, D. I., Bosch, J., and Olsson, H. H. (2020). Data labeling: An empirical investigation into industrial challenges and mitigation strategies. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12562 LNCS:202–216. doi:10.1007/978-3-030-64148-1_13/TABLES/1.
- Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., and Rodríguez, J. G. (2017). A review on deep learning techniques applied to semantic segmentation. *CoRR*, abs/1704.06857. doi:10.48550/arXiv.1704.06857.
- Gu, H., Wang, Y., Hong, S., and Gui, G. (2019). Blind channel identification aided generalized automatic modulation recognition based on deep learning. *IEEE Access*, 7:110722–110729. doi:10.1109/ACCESS.2019.2934354.
- Hossain, A. and Sajib, S. A. (2019). Classification of image using convolutional neural network (cnn). *Global Journal of Computer Science and Technology*, 19:13–18. doi:10.17406/GJCST.
- Kadunc, N. O. (2022). How to normalize satellite images for deep learning. URL <https://medium.com/sentinel-hub/how-to-normalize-satellite-images-for-deep-learning-d5b668c885af>.

- Khan, A., Chatterjee, S., and Weng, Y. (2021). 2 - characterizing thermal fields and evaluating uhi effects. In Khan, A., Chatterjee, S., and Weng, Y., editors, *Urban Heat Island Modeling for Tropical Climates*, pages 37–67. Elsevier. doi:10.1016/B978-0-12-819669-4.00002-7.
- Liu, H., Zhan, Q., Yang, C., and Wang, J. (2018). Characterizing the spatio-temporal pattern of land surface temperature through time series clustering: Based on the latent pattern and morphology. *Remote Sensing*, 10(4). doi:10.3390/rs10040654.
- Long, J., Shelhamer, E., and Darrell, T. (2014). Fully convolutional networks for semantic segmentation. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440. doi:10.48550/arXiv.1411.4038.
- Lotfian, M., Ingensand, J., and Composto, S. (2019). The relationship between land surface temperature and local climate zone classification : A case study of the canton geneva , switzerland. URL <https://api.semanticscholar.org/CorpusID:199514954>.
- O’Shea, K. and Nash, R. (2015). An introduction to convolutional neural networks. *International Journal for Research in Applied Science and Engineering Technology*, 10:943–947. doi:10.22214/ijraset.2022.47789.
- Pedrayes, O. D., Lema, D. G., García, D. F., Usamentiaga, R., and Alonso, (2021). Evaluation of semantic segmentation methods for land use with spectral imaging using sentinel-2 and pnoa imagery. *Remote Sensing*, 13(12). doi:10.3390/rs13122292.
- Prakash, A. (2000). Thermal remote sensing: concepts, issues and applications. *International Archives of Photogrammetry and Remote Sensing*, 33(B1; PART 1):239–243.
- Ren, Z., Fu, Y., Dong, Y., Zhang, P., and He, X. (2022). Rapid urbanization and climate change significantly contribute to worsening urban human thermal comfort: A national 183-city, 26-year study in china. *Urban Climate*, 43:101154. doi:https://doi.org/10.1016/j.uclim.2022.101154.
- Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597. doi:10.48550/arXiv.1505.04597.
- Singh, S., Mall, R. K., Chaturvedi, A., Singh, N., and Srivastava, P. K. (2024). Advances in remote sensing in measuring urban heat island effect and its management. *Earth Observation in Urban Monitoring: Techniques and Challenges*, pages 113–132. doi:10.1016/B978-0-323-99164-3.00011-2.
- Stewart, I. D. and Oke, T. R. (2012). Local climate zones for urban temperature studies. *Bulletin of the American Meteorological Society*, 93:1879–1900. doi:10.1175/BAMS-D-11-00019.1.
- Suzuki, K. (2013). Artificial neural networks - architectures and applications. *Artificial Neural Networks - Architectures and Applications*. doi:10.5772/3409.
- Xu, G., Zhu, X., Tapper, N., and Bechtel, B. (2019). Urban climate zone classification using convolutional neural network and ground-level images. *Progress in Physical Geography*, 43:410–424. doi:10.1177/0309133319837711/ASSET/IMAGES/10.1177_0309133319837711 - IMG18.PNG.
- Zhang, D., Wang, J., and Zhao, X. (2015). Estimating the uncertainty of average f1 scores. In *Proceedings of the 2015 International Conference on The Theory of Information Retrieval, ICTIR '15*, page 317–320, New York, NY, USA. Association for Computing Machinery. doi:10.1145/2808194.2809488.

Bibliography

- Zhang, Y., Li, Y., Chen, Y., Liu, S., and Yang, Q. (2022). Spatiotemporal heterogeneity of urban land expansion and urban population growth under new urbanization: A case study of chongqing. *International Journal of Environmental Research and Public Health* 2022, Vol. 19, Page 7792, 19:7792. doi:10.3390/IJERPH19137792.
- Zhao, Z., Sharifi, A., Dong, X., Shen, L., and He, B. J. (2021). Spatial variability and temporal heterogeneity of surface urban heat island patterns and the suitability of local climate zones for land surface temperature characterization. *Remote Sensing* 2021, Vol. 13, Page 4338, 13:4338. doi:10.3390/RS13214338.
- Zheng, Y., Ren, C., Xu, Y., Wang, R., Ho, J., Lau, K., and Ng, E. (2018). Gis-based mapping of local climate zone in the high-density city of hong kong. *Urban Climate*, 24:419–448. doi:10.1016/J.UCLIM.2017.05.008.

Colophon

This document was typeset using L^AT_EX, using the KOMA-Script class scrbook. The main font is Palatino.

