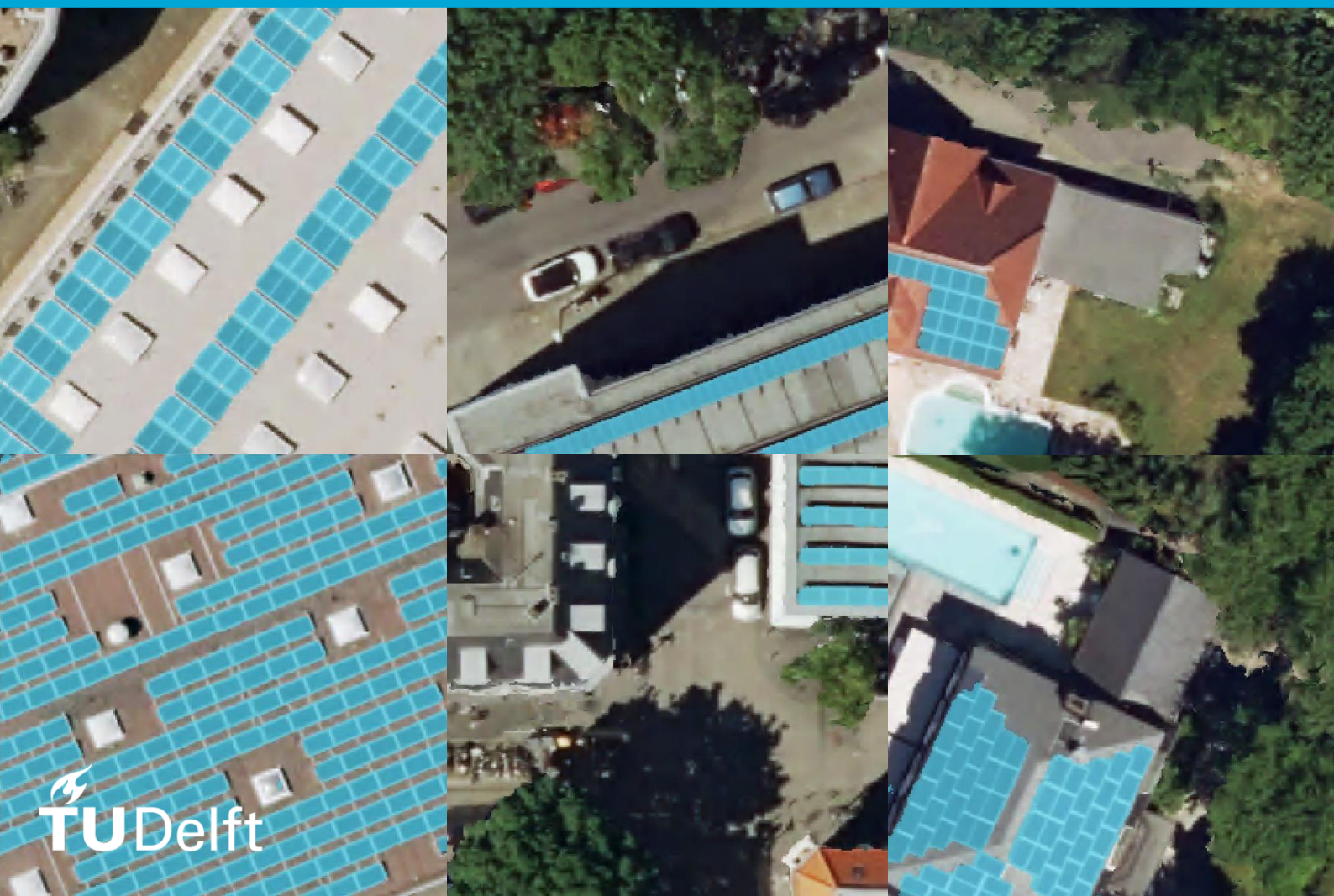
MSc thesis in Geomatics for the Built Environment

Automated rooftop solar panel detection through Convolutional Neural Networks

Simon Pena Pereira

2023



TU Delft

MSc thesis in Geomatics

Automated rooftop solar panel detection through Convolutional Neural Networks

Simon Pena Pereira

January 2023

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

The work in this thesis was carried out in the:

Geo-Database Management Center
Delft University of Technology

Supervisors:	Dr. Azarakhsh Rafiee
	Dr. Stef Lhermitte
Co-reader:	Dr. Roderik Lindenbergh

Abstract

Transforming the global energy sector from fossil-fuel based to renewable energy sources is key to limiting global warming and efficiently achieving climate neutrality. The decentralized nature of the renewable energy system allows private households to install photovoltaic (PV) systems on their rooftops. In this context, planning an efficient grid expansion is becoming increasingly difficult. Therefore, deep learning (DL) techniques, such as convolutional neural networks (CNNs), can support collecting meta data about PV systems from aerial or satellite images, as research in the field of remote sensing has shown. However, previous research lacks the consideration of ground truth data-specific characteristics of PV panels.

This thesis aims to implement a semantic segmentation model that detects PV systems in aerial imagery to emphasize the relevance of area-specific characteristics for the training data and convolutional neural network (CNN) hyperparameters. A CNN with U-Net architecture is employed to analyze the impacts of land use types, rooftop colors, near-infrared (NIR) data, and lower-resolution images on the detection rate of PV panels in aerial imagery. The results indicate that a U-Net is suitable for classifying PV panels in high-resolution aerial images (10 cm) by reaching F1-scores of up to 91.75% while demonstrating the importance of adapting the training data to area-specific ground truth data in terms of urban and architectural properties.

Keywords: PV panels, CNN, U-Net, semantic segmentation, aerial imagery

Acknowledgements

First, I would like to thank my thesis supervisor Dr. Azarakhsh Rafiee for the valuable discussions, guidance, and support throughout my thesis. Also, I would like to thank my second supervisor Dr. Stef Lhermitte for always providing valuable feedback, which helped me to steer in the right direction, and my co-reader Dr. Roderik Lindenbergh for his feedback in the final period of my thesis.

Lastly, I would like to express my deepest gratitude to my parents for their relentless support and encouragement throughout the years, as well as to Maren and Sophia for their patience and confidence in me and for pushing me through my studies.

Contents

List of Figures

List of Tables

Acronyms

1. Introduction

Nearly three-quarters of human-caused greenhouse gas emissions that drive climate change stem from the energy sector, making climate change primarily an energy problem (Climate-Watch, 2022). Therefore, the energy sector is increasingly shifting towards more renewable and sustainable energy sources in line with the Paris Agreement commitments to limit global warming to an average of well below 2°C compared to the pre-industrial level (UNFCCC, 2015). Transitioning to renewable energy technologies is key to a clean and secure energy system on the path to climate neutrality (UN, 2022). In this transition, solar energy is the fastest-growing and most competitive source of renewable energy in the European Union (EC, 2022).

Popular technologies to convert sunlight into energy are PV systems, concentrated solar power systems, and solar thermal systems. Both, PV systems and concentrated solar power systems convert solar energy into electricity. While concentrated solar power systems are large-scale systems using mirrors that concentrate sunlight towards a receiver generating heat to power steam turbines, PV systems rely on solar cells using the photovoltaic effect. These cells compose PV panels that can be installed in large-scale solar power plants on the ground or as floating PV systems on lakes but also in form of decentralized PV systems on rooftops. In contrast, solar thermal systems mainly generate heat to produce hot water for residential buildings (EC, 2022).

The energy sector's growth is expected to continue in the upcoming decades mainly driven by PV systems which are the most accessible sources of renewable energy for private households (EC, 2022). Due to this liberalization of the energy sector, national agencies, such as the Federal Network Agency of Germany, demand a comprehensive and reliable data basis for planning grid expansions (MaStR, 2023). Depending on the country, well to poorly-documented registries of active PV systems exist, which are a hurdle for decision makers involved in the development of an efficient energy transition.

An alternative method for populating the registries with up-to-date information about installed PV systems is the use of deep learning algorithms that learn how to detect objects in satellite or aerial imagery. As indicated by Rausch et al. (2020), deep learning algorithms for image classification, such as CNNs, can be useful for validating, updating, and completing PV system registries.

1.1. Motivation

The comprehensive work by De Jong et al. (2020) demonstrates the ability to classify solar panels with CNNs. To optimize the effectiveness as well as the efficiency of these algorithms, research has mainly focused on the technical configurations of these networks. However, the performance of these algorithms is calculated on the basis of the network's prediction in comparison to the ground truth data. Therefore, it is also crucial to understand the

impact of diverse ground truth data on the performance of the network. The importance to analyze the ground truth of PV panels and their surrounding with regard to differences in land use or architectural characteristics becomes evident when statistics about PV panels are created on a national or international scale. This hurdle became evident when De Jong et al. (2020) conducted validations across different geographical areas. As regions or countries can differ in building densities and sizes, rooftop colors and shapes, and sizes of PV systems, it is of great relevance to know the impact of different aspects on the classification process. Moreover, Da Costa et al. (2021) demands a shift from model-driven to data-driven research to detect PV panels. By comparing multiple CNN models, it became evident that their results differ insignificantly, which highlighted the importance of the reliable and comprehensive data sets of annotated PV panels. Knowing the ground truth characteristics helps to collect appropriate ground truth data and to adapt the algorithm in such a way that it can compute reasonable predictions for the object of interest.

1.2. Objective and scope

This thesis aims to emphasize the importance of considering variations in ground truth data when utilizing DL networks for the classification of PV panels in diverse urban environments. To meet this objective, it requires a data pipeline that includes high-resolution aerial imagery, area-specific ground truth data of PV panels, an appropriate algorithm for detecting PV panels, and different methods for assessing the algorithm's performance.

Pointing out specific causes that counteract or reinforce an effective detection of PV panels is hardly possible when too many urban properties are included in the ground truth data. To narrow down the variation of urban characteristics, the ground truth data collection is limited to local areas within the city of Cologne in Germany. In this way, impacts on the detection process can be isolated efficiently. Furthermore, additional and modified data in form of NIR image channels and lower-resolution images are incorporated into the research to allow conclusions concerning additional spectral information and different spatial resolutions.

The DL algorithm employed in this study is a CNN with a U-Net architecture developed by Ronneberger et al. (2015). The U-Net architecture is a straightforward CNN that has demonstrated promising results for similar applications in previous research (Castello et al., 2019; Da Costa et al., 2021). It computes semantic segmentations which are classified images in which each pixel is associated with a target class or background information. Minor modifications in the U-Net are required to obtain semantic segmentation with the same dimensions as the input image and label. The input consists of manually generated ground truth labels representing the target class of PV panels and aerial images at a resolution of 10 cm per pixel to compute pixel-based classifications of PV panels. Overall, the methodologies employed in this thesis are compiled into a semi-automated pipeline, meaning that manual working steps are required in the pre-and post-processing stages to execute the pipeline from end to end. Further networks are not considered since a variation of CNNs does not contribute to the previously defined objectives.

1.3. Research questions

Following the objectives and the scope outlined in Section 1.2, this subchapter defines one main research question and four subquestions that meet both conditions. The main question covers the core of this thesis which is the application of the U-Net architecture to classify PV panels on rooftops. It is defined as follows:

To what extent is a CNN with U-Net architecture suitable for detecting PV panels on rooftops in aerial images?

The following four subquestions analyze the main question from different perspective:

- What is the impact of different land use types on the detection of PV panels?
- What is the effect of adding near-infrared data to aerial images on the detection of PV panels?
- How is the correlation between roof and panel color affecting the detection of PV panels?
- How sensitive is the model towards lower resolutions with regard to the panel size?

1.4. Thesis outline

The thesis document is structured into 6 chapters. Following the introduction, the theoretical background knowledge is presented in Chapter 2 that is needed to understand the implementation of the U-Net as well as related research outcomes. In Chapter 3, an overview of the employed methodology is given to outline each working step of collecting data, preprocessing data, and analyzing the results. Based on this, the technical implementation of the methodology is explained in detail in Chapter 4. Furthermore, it defines the study area and the data used for the implementation. Moreover, the technical modifications of the employed U-Net are described. Following this, the results of the model are summarized and analyzed in Chapter 5. In the final chapter (Chapter 6), the results are summarized and discussed by answering the research questions. Additionally, the contribution of this thesis to current research is described as well as suggestions for potential future work.

2. Theoretical background and related work

This chapter gives an introduction to deep learning and explains the mathematical background of the neural network implemented for this thesis. Additionally, it provides an overview of research projects dealing with DL algorithms to detect PV panels on aerial or satellite images.

2.1. Deep learning

DL is often considered a sub-field of machine learning (ML) which originates in the field of artificial intelligence (AI). DL models are composed of multiple layers that process nonlinear information to learn representations of data at multiple levels of abstraction (Deng and Yu, 2014; Lecun et al., 2015). Due to increased chip processing abilities, e.g., with parallel computing on graphical processing units (GPUs), the increase of available training data, and the advances of ML research, the DL-field has become increasingly popular (Deng and Yu, 2014). Especially with regard to real-world applications, such as image and human speech recognition, DL methods were able to outperform the limited capabilities of conventional ML methods (Lecun et al., 2015).
The wide range of DL techniques can be categorized into three main classes, namely deep networks (DNs) for unsupervised learning, DNs for supervised learning, and hybrid DNs. The key aspect of unsupervised learning is the absence of information about the target class labels in the learning process. The algorithm aims to capture high-order correlation to analyze patterns in data. In contrast, supervised learning algorithms are capable of providing discriminative power for pattern recognition due to always available information on target class labels. The combination of both categories leads to hybrid DNs that can make use of supervised DNs assisted by the outcome of unsupervised DNs, or vice versa where the discriminative power helps to estimate the parameters of unsupervised DNs (Deng and Yu, 2014).
The DN used in this thesis is a CNN which is based on artificial neural networks (ANNs) for supervised learning.

2.2. Artificial Neural Network (ANN)

ANNs are computational processing systems inspired by biological nervous systems (O'Shea and Nash, 2015). Similar to the human brain which is composed of interconnected neurons, an artificial neural network (ANN) consists of simple processing elements, also referred to as nodes or units (O'Shea and Nash, 2015; Bishop, 1998). According to the all-or-none law, neurons *fire* an electrical impulse (also referred to as action potential) along the axon to the

synapses of the next neuron. The magnitude of the impulse's effect depends on the neuron's strength, which is in terms of ANNs analogous to the parameter *weight* or w_i of a unit. At each unit, the weight is multiplied with an incoming signal x_i at input i. If the weighted sum of inputs from other units exceeds a certain threshold, the unit *fires* a signal to the next unit. The firing threshold corresponds to the constant weight w_0 which is called *bias* (Bishop, 1998). It corresponds to y-intercept of a linear equation (see Equation 2.1).

$$a = \sum_{i=1}^{d} w_i x_i + w_0, \tag{2.1}$$

The output of a single processing unit z is given from processing a in a non-linear activation function $g()$ (see Equation 2.2) which is described in more detail in Section 2.2.3.

$$z = g(a). \tag{2.2}$$

One of the most common classes for a feedforward ANN is the multilayer perceptron (MLP) which processes a multidimensional input vector through a set of *hidden units*, a so-called *hidden layer*, in which the weighted sums of multiple units determine the final output (Figure 2.1). These decisions made in the hidden layer are associated with the process of learning. By stacking multiple hidden layers upon each other, a certain depth is given to the network which is commonly called deep learning or deep neural network (DNN) (O'Shea and Nash, 2015).

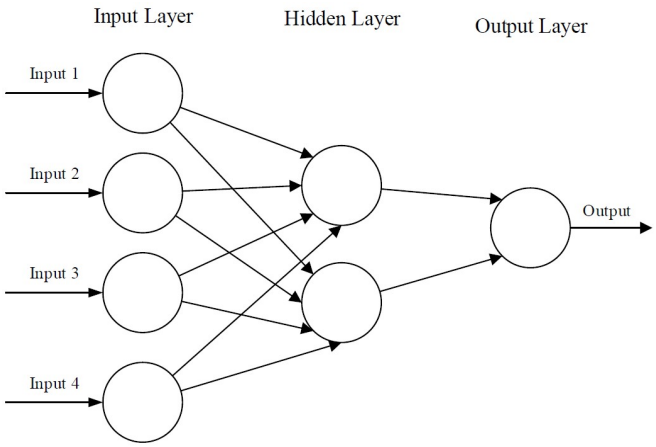


Figure 2.1.: Feedforward neural network (O'Shea and Nash, 2015)

A popular learning method that is complementary to the learning process of feedforward multilayer perceptrons (MLPs) is backpropagation (Lecun et al., 2015). Its objective is the adjustment of weights at all units to minimize the measured difference between the predicted output of the network and the desired output which corresponds to the target class labels the algorithm is fed with. By doing that, the network can self-organize its weights to construct appropriate internal representations for a specific task (Rumelhart et al., 1986). The key aspect is the computation of the gradient (or derivative) of an objective function (described in more detail in Section 2.2.4) with respect to all weights in the network. This

is achieved by a backward pass through the network from the output layer at the top to the input layer at the bottom (Lecun et al., 2015). This procedure minimizes the total error of the network's performance to reach a closer distance between the predicted output and the label class target. The cycle of both a forward pass and a backward pass of data through the network is called an epoch. For an appropriate adjustment of the weights, multiple epochs can be executed consecutively.

In this thesis, an ANN class called CNN is applied for semantic segmentation of multispectral orthophotos (described in Section 2.3). To run an ANN or CNN several decisions need to be made by the user on which functions to incorporate into the network.

2.2.1. Input data split

The concept of a data split lies in constructing a network that generates representative and unbiased predictions of data. Evaluating the network's performance within and after the training process helps the user to choose the appropriate network. Furthermore, it avoids the selection of a network that is overfitted to the training data, making it unsuitable for predictions on an independent testing set. Therefore, splitting the data set in training, validation, and testing data set is a common approach to assess the network with validation metrics (described in Section 2.6).

Typically, most data is assigned to the training data set that propagates through the network. An important approach to evaluate the network's performance in the training process is the use of an independent validation data set that was not seen by the network before. By doing that, the training progress can be observed based on an evaluation after each epoch. Lastly, the network's overall performance can be assessed on the prediction of the testing data set. Further, splitting the data with random sampling is the preferred approach to form subsets (Foody, 2017).

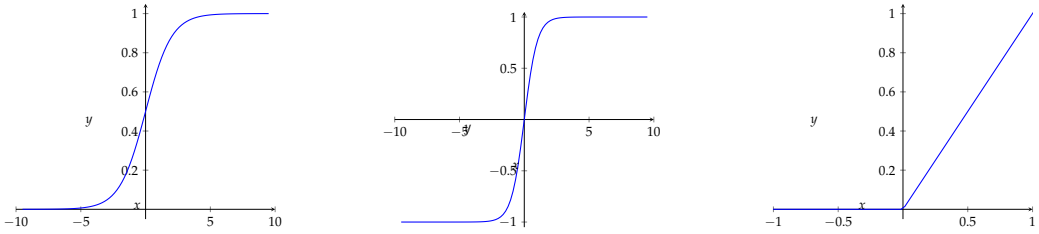
2.2.2. Weight and bias initialization

Training an ANN for the first time requires the initialization of weights at each unit so they can then be modified through backpropagation as well as the initialization of a constant bias. A common method for CNNs is a random sampling from the Gaussian distribution, e.g., with a mean value of *0*, a fixed standard deviation of 0.01, and a bias of 0 (Zeiler and Fergus, 2013; Simonyan and Zisserman, 2014). However, this approach slows down the learning process and may cause a poorer local optimum (He et al., 2015; Xu and Wang, 2022).

An improved method called *Xavier* initialization was presented by Glorot and Bengio (2010) that consists of a uniform distribution *U[-a,a]* with a mean of 0 and a variance of $\frac{1}{n}$ where n is the number of input features at each unit (Xu and Wang, 2022). Similar to the *Xavier* initialization, He et al. (2015) proposed a method with a mean value of 0 and a variance of $\frac{2}{n}$ that is specifically designed for the use of the ReLU activation function.

2.2.3. Activation function

As indicated by Equation 2.2, an ANN requires an activation function at each unit to process a (see Equation 2.1) of one unit to the next unit in the network. Three typical activation functions for ANNs are presented in Figure 2.2 below.



((a)) ((b)) ((c))

Figure 2.2.: (a) sigmoid (b) tanh (c) ReLU

An often implemented activation function for ANNs was the tanh function which lost in popularity due to ReLU that allows faster learning processes in deep ANNs (Lecun et al., 2015). Furthermore, ReLU is easier to optimize and it demonstrates greater abilities in generalization (Zeiler and Fergus, 2013).

The output of ReLU is either 0 or the input itself x (see Equation 2.3). If the input is smaller or equal to 0 then the activation function does not $fire$ or pass the output to the next unit. In order to move the threshold of the activation function, a bias unequal to 0 can shift the activation function in both directions along the x-axis. Consequently, the value of the bias corresponds to the opposite of the activation function's threshold.

$$ReLU(x) = max(0, x) \tag{2.3}$$

Commonly, either sigmoid or softmax functions are implemented in the output layer. While softmax functions are typically used for multiclass classification problems, the sigmoid $\sigma(x)$ function is used for binary classifications to provide a probability classification between 0 and 1 (Equation 2.4) (Sharma et al., 2020).

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{2.4}$$

2.2.4. Loss and cost function

As described in Section 2.2, a crucial part of the learning process is the ability to train parameters through backpropagation. To define an appropriate adjustment of weights and biases, the optimizer requires knowledge about the training progress at the end of each feedforward propagation. Therefore, the loss function calculates the difference between the predicted value $\hat{y}$ and the target class label y to provide feedback to the optimizer (see following Section 2.2.5) that adjusts the weights. In the literature, the term cost function is often used interchangeably with the term loss function, however, some authors are distinguishing

between the loss function assessing the difference of $\widehat{y}$ and y and the cost function averaging over all training examples (Bottou, 1991).

A simple and often used loss function in neural networks is the mean squared error (MSE). Giving its name, it calculates the mean of the squares of the errors between predicted and observed values (see Equation 2.5).

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2 \tag{2.5}$$

A widely used loss function for binary classification is the BCE (see Equation 2.6). For multi-class classifications with more than one target class, the categorical cross-entropy loss function can be applied.

$$BCE = \frac{1}{n}\sum_{i=1}^{n}(y_i \log(\widehat{y}_i) + (1 - y_i)\log(1 - \widehat{y}_i)) \tag{2.6}$$

Built on the cross entropy loss function, Lin et al. (2017) proposed a dynamically scaled alternative called FL (see Equation 2.7). It is specifically designed to tackle extreme imbalances between the target class label and the background class. A so-called *focusing* parameter γ is implemented that yields the best results at a value of 2 according to the experiments of Lin et al. (2017).

$$FL(p_i) = -(1 - \widehat{y}_i)_\gamma \log(\widehat{y}_i) \tag{2.7}$$

2.2.5. Optimizer

The most common method to optimize ANNs is the implementation of a gradient descent (GD) algorithm. It is based on a convex function (see Figure 2.3) and aims to minimize the loss function by determining the appropriate parameters. This is done in the opposite direction of the gradient of the loss function. The minimum loss for appropriate weight values is defined by the lowest point of the convex function, called the point of convergence. The step size taken to reach the convergence is the learning rate. Usually, the learning rate is defined as very small, although defining it too small causes a slow convergence while a large learning rate results in a loss function fluctuating around the point of convergence (Ruder, 2016).

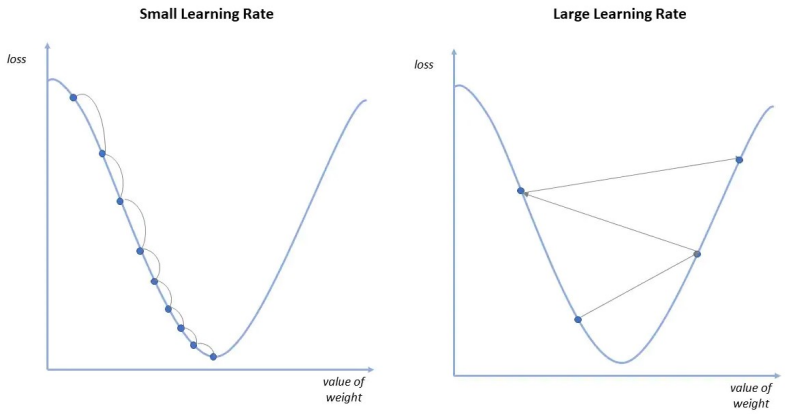


Figure 2.3.: Comparison of small and large learning rates approaching the convergence (IBM, 2020)

There are three ways to incorporate the gradient descent in the training process, namely by the batch gradient descent (BGD) (or vanilla GD), the stochastic gradient descent (SGD), and the mini-batch GD (Ruder, 2016). The BGD computes the gradient of the loss function for the entire data set so that the parameters are only adjusted once. In contrast to that, SGD updates the parameters for one training example at a time. The combination of BGD and SGD yields the method of mini-batch GD, where training examples are bundled up in batches for which the parameters are adjusted respectively.

An efficient and robust algorithm for gradient-based optimization is the adaptive moment estimation (Adam) proposed by Kingma and Ba (2014). It is used as a basis for many other optimizers and has proven to outperform others. Further approaches to optimize the SGD with Adam are regularization methods such as batch normalization and the early stopping described in Section 2.5.1 and Section 2.5.2 (Ruder, 2016).

2.3. Convolutional Neural Network

The roots of CNNs lie in the *necognitron* introduced by Fukushima (1980), building the foundation for one of the most popular publications about CNNs by Lecun et al. (1998) which deals with the automated recognition of hand-written digits.

A CNN is a deep network (DN) with a hierarchical structure to extract high-level semantic information from images by recognizing patterns (Alam et al., 2021). It is superior to a traditional ANN which struggles with the complexity of image data since it considers each pixel as an input neuron with its own weight and bias. The high number of trainable parameters causes a complexity that is both highly limited to computational power as well as prone to overfit to training data. Overfitting is described as the reduced ability to generalize abstract information since the model is learning too close to the training data set but is unable to do prediction of an unseen test data set. The reduced number of trainable parameters allows

the computation of much more complex input data which makes it specifically suitable for processing images (O'Shea and Nash, 2015).

In general, a CNN model consists of a series of stages that are composed of up to three different layers, namely a convolutional layer with elementwise activation functions (see Section 2.2.3), a pooling layer, and fully connected layers with implemented activation functions for classification (see Figure 2.4) (O'Shea and Nash, 2015; Lecun et al., 2015). The batch normalization is an optional layer for regularization purposes that is described in more detail in Section 2.5.1.

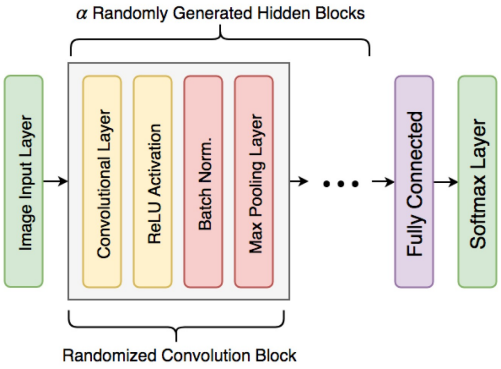


Figure 2.4.: Typical structure of a convolutional block (Wang et al., 2019)

2.3.1. Convolutional layer

In a convolutional layer, a filter matrix (or kernel) consisting of weights is sliding across an input image to compute the dot product with the pixel values of local patches (see Figure 2.5). Each patch is denoted as a receptive field of an unit in the output layer. The output matrix is a feature map (or activation map) that serves as an input to the following convolutional layer (O'Shea and Nash, 2015). Lecun et al. (1998) describe the local connections between feature maps and the corresponding weight sharing at each unit as a key aspect of CNNs. The set of weights is the same for all units of a feature map, so it can detect the same pattern in all parts of an image. In order to detect different types of patterns, each feature map is provided with a different set of filter weights (Lecun et al., 2015).

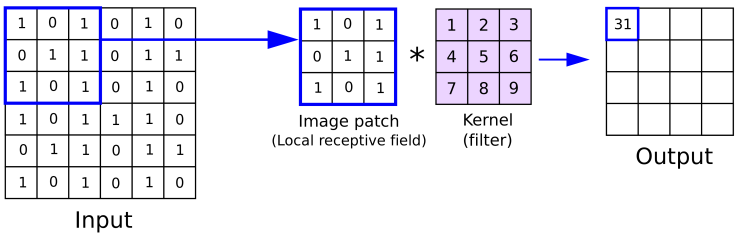


Figure 2.5.: Convolutional operation (Reynolds, 2019)

Each convolutional operation shrinks the size of the feature map (see Figure 2.5). Therefore, a spatial padding operation called zero-padding is applied to preserve the same spatial resolution. This is done by adding zero values to the feature map's border. This method only works with a convolutional stride of 1 (Simonyan and Zisserman, 2014).

The stride is defined as the distance between the centers of two filters that belong to neighboring units in the feature map (Krizhevsky et al., 2017). In other words, it determines the number of pixels the filter moves along an input image.

2.3.2. Pooling layer

Another key aspect of CNNs are pooling operations that aim to merge similar semantic information of feature maps into a new feature map. Typically, a max-pooling layer is incorporated at the end of each convolutional block as shown in Figure 2.4. It computes the maximum of local patches of units which are outputted in a new feature map (see Figure 2.6). As in convolutional operations, the feature maps shrink in dimensions. However, this is intended for the max-pooling operation (Lecun et al., 2015). It furthermore reduces the number of trainable parameters and therefore the complexity of the model. Commonly, a kernel size of $2x2$ with a stride of 2 is applied (O'Shea and Nash, 2015).

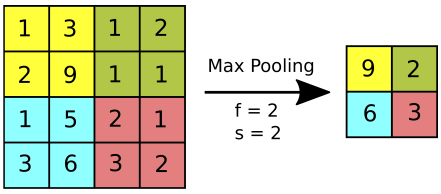


Figure 2.6.: Max-pooling operation (Reynolds, 2019)

2.3.3. Fully connected layer

The last layer before the output layer is the fully connected layer, which connects all units with the units of the next layer. It inherits an activation function for computing the classification (O'Shea and Nash, 2015).

2.4. U-Net architecture

The network architecture that is going to be used in this thesis was presented by Ronneberger et al. (2015) and is called $U-Net$ since it is shaped like the letter U.

The U-Net is based on the architecture of the so-called fully convolutional network. It is modified in such a way that it allows training with few input images. Instead of using fully connected layers it relies on the valid part of each convolution which denotes the output of an unpadded (see *padding* in Section 2.3.1) convolution. As illustrated in Figure 2.7, it consists of a contracting path (or encoder) on the left side, where the input is downsampled by max-pooling layers, and an expansive path (or decoder) on the right side, where upsampling operations are carried out by up-convolutional layers. Overall, it is composed of four

convolutional blocks on each side, in addition to the convolutional block at the bottom of the U-Net which receives downsampled input and outputs uplsampled layers (Ronneberger et al., 2015).

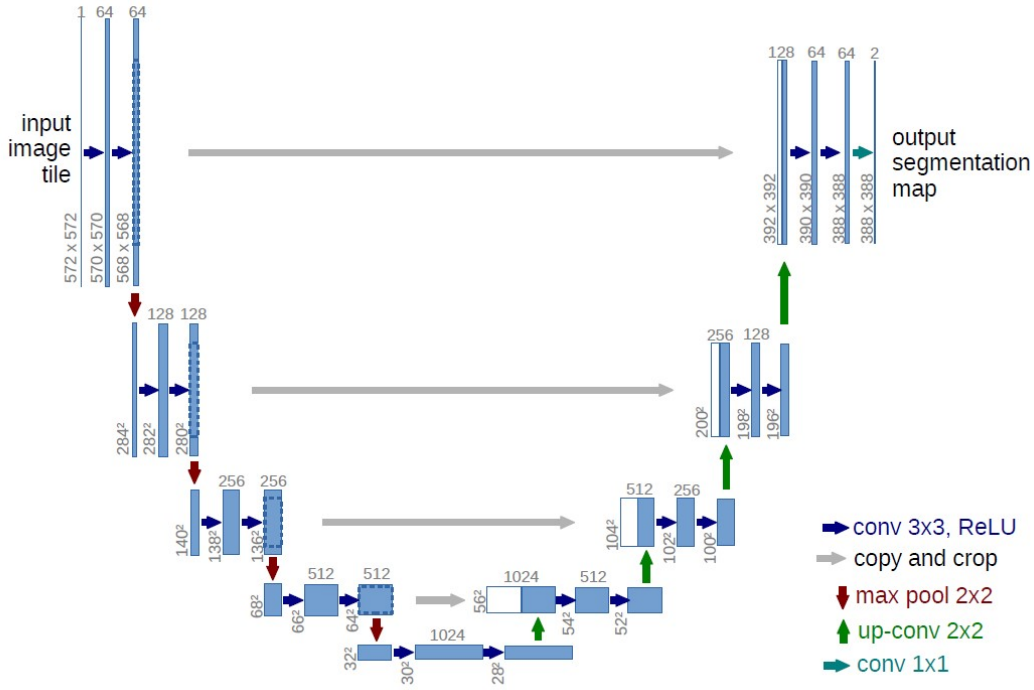


Figure 2.7.: U-Net architecture consisting of a contracting and an expansive path with multiple convolutional and pooling operations applied on multi-channel feature maps as blue boxes (Ronneberger et al., 2015)

In more detail, each convolutional block in the contracting path is composed of two consecutive 3x3 convolutional layers, followed by a ReLU activation function, and a 2x2 max-pooling layer with a stride of 2. The convolutional layers in the first block contain 64 filter layers outputting a multi-channel feature map of 64 feature layers. Due to a convolutional stride of 1, the input layer size decreases by 1 row or column of pixels at each side of the input layer. Accordingly, both the x- and y-sizes of a layer shrink by the value of 2 after each convolutional operation. With each max-pooling layer (with a stride of 2) the dimensions of the layers in the next convolutional block are halved while the number of feature maps is doubled. Localizing the feature representation computed in the contracting path requires an increase in resolution which is achieved with four consecutive up-convolutional operations (or transposed convolutions) in the expansive path. The higher resolutions of feature maps are reconstructed by cropping and copying the feature maps from the respective level of the contracting path. The convolutional layers in the expansive path follow the same structure as in the encoding process, except for an additional final layer at the end of the U-Net that is applying a 1x1 convolution to map the classification.

2.5. Regularization

In general, regularization methods intend to prevent the model from overfitting. In the following, two methods are presented that can regularize the training process of a CNN.

2.5.1. Batch normalization

Applying SGD optimization (as described in Section 2.2.5) is an effective way of training a model. Nevertheless, it requires a careful selection of the model's hyper-parameters due to its effect on the input data that amplifies with the increasing depth of the network. In addition, the data distribution can change depending on the variation of input values. The resulting change of unit distributions is referred to as *internal covariate shift*. To address this issue, Ioffe and Szegedy (2015) proposed a mechanism called *batch normalization* that can be implemented in a CNN structure to make the model more robust and to increase the computational speed of training by allowing higher learning rates. The idea behind batch normalization is to limit the distribution shifts of output values at each hidden layer. This is achieved by fixing the means and variances of feature maps which cause a better gradient flow through the CNN. In the context of batches, this is implemented by constraining the output values of each feature map of a batch to have the same mean and variance. Also, this allows the user to be less careful about parameter initialization.

2.5.2. Early stopping

Early stopping is a regularization method to stop the optimization process of the GD at an appropriate point in the training process to avoid overfitting and underfitting (see Figure 2.8) (Zhang and Yu, 2005). This point in time is reached when there is no further improvement in the validation loss observed, therefore, a stagnation of the loss function over a certain number of epochs. When the stagnation is exceeded, the validation loss increases again since the model learned too close to the training data set and it is not able to classify the validation data set. This method helps the user approximate an ideal number of epochs and saves computational time.

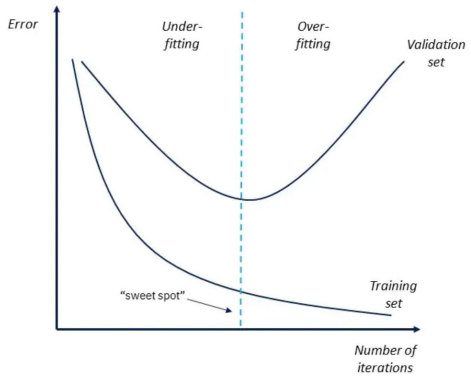


Figure 2.8.: Training and validation loss with increasing number of epochs; Early stopping at sweet point (IBM, 2021)

2.6. Model evaluation

2.6.1. Accuracy

The most common metric is accuracy which is defined by the number of correctly predicted images divided by the total number of predictions. Considering the context of this thesis, correctly predicted images include true positives (TP), in which PV panels are correctly identified as well as true negatives (TN) (absence of PV panels is correctly identified).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.8}$$

2.6.2. Precision

Another metric called precision calculates the proportion of true positives to the total number of actual PV panels, including TP and not identified PV panels or false positives (FP).

$$Precision = \frac{TP}{TP + FP} \tag{2.9}$$

2.6.3. Recall

To express the proportion of correctly identified PV panels to all predictions of PV panels, the recall metric will be applied. It is calculated by the number of TP divided by the number of TP and FN.

$$Recall = \frac{TP}{TP + FN} \tag{2.10}$$

2.6.4. F1-score

Further, there is the F1-score which expresses the harmonic average of precision and recall. It computes the overlap between ground truth data and prediction and divides it by the total number of pixels.

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * TP}{2 * TP + FP + FN} \tag{2.11}$$

2.6.5. Intersection over Union (IoU) - Jaccard index

The Jaccard index or intersection over union (IoU) is a coefficient to measure the similarity between two samples. It is calculated by dividing the intersection between the label and prediction by the union of both samples. Its formula is similar to the F1-score, which is why both validation metrics are positively correlated. Both output scores range between 0 and 1. For the IoU, 0 indicates no overlap between ground truth data and prediction, while a score of 1 represents a total overlap. In comparison, the IoU penalizes under- and over-segmentation more than the F1-score, which is based on the greater impact of FN and FP (Müller et al., 2022).

$$Jaccard(U, V) = \frac{|U \cap V|}{|U \cup V|} = \frac{TP}{TP + FP + FN} \tag{2.12}$$

2.7. Deep Learning in remote sensing

Due to the increase in computing power in recent years, more and more deep neural networks were presented to solve object detection tasks in computer vision, and thereafter in the remote sensing domain. Building on that knowledge, many CNN architectures have been implemented in remote sensing analysis of the urban environment, for instance, for the purpose of detecting roads, buildings, or vehicles (Shi et al., 2017; Ševo and Avramović, 2016; Vakalopoulou et al., 2015; Chen et al., 2014). Due to the massive scale-up of renewable energy, new applications for remote sensing techniques emerged in the solar energy domain, such as the evaluation of the PV potentials of building rooftops (Chen et al., 2022), the detection of damaged PV panels (Pierdicca et al., 2018), or the localization of PV panels (De Jong et al., 2020; Da Costa et al., 2021; Rausch et al., 2020; Castello et al., 2019; Malof et al., 2017).

In the following two sections, relevant literature concerning the detection of PV panels in aerial images with CNNs is presented. First, Section 2.7.1 outlines a comprehensive project that conducted image classification and object detection methods to identify PV panels in aerial images. Section Section 2.7.2 introduces the results of three research projects that carried out semantic segmentation, which is of great relevance to this thesis. Lastly, a brief overview of PV panel types is given as PV panels are the objects of interest (see Section 2.7.3).

2.7.1. Convolutional Neural Networks for Image Classification and Object Detection

This section introduces the DeepSolaris project that was carried out by four national statistical offices from the Netherlands, Germany, and Belgium, together with the Open Universiteit Nederland (De Jong et al., 2020). It is especially interesting as a reference to this thesis since it includes the German state of North Rhine-Westphalia (NRW) as a study area which partially covers the area of interest (AOI) for this thesis and therefore uses similar data sets.

Prior to the DeepSolaris publication, Curier et al. (2018) published a corresponding article describing the data, its pre-processing steps as well as the annotation process for creating

label data in more detail. These are crucial information for creating a suitable input data set for this thesis.

Besides the state of NRW, the algorithms in the DeepSolaris project were also trained on the province of Limburg in the Netherlands. The project aimed to create a new way of producing official statistics by retrieving information from aerial images with deep learning algorithms. The main objective was to map the locations of PV panels at a regional to a local level and, thus provide a better understanding of the energy transition in the context of PV panel installations. To achieve this goal, two approaches were tested for the state of NRW. Firstly, the approach of image classification, in which the algorithm predicts whether the image contains a PV panel or not. This approach demands a different method of annotating the images because it does not determine the precise PV panel location within the image but the image itself. They applied both models, InceptionResNetV2, as well as VGG16, with pre-trained weights from ImageNet, which turned out to be beneficial for the performance of both networks. Secondly, the approach of object detection is divided into two stages. While the first stage is proposing potential pixel regions for the presence of an object, the second stage is localizing the object. The most common localization method is the use of bounding box regressions to predict the object's exact location. For this approach, they went one step further by applying the Mask R-CNN algorithm that computes pixel-based masks of the object, in addition to the bounding box.

Overall, the DeepSolaris project demonstrated the ability of CNNs to detect PV panels in an almost automatic manner. It furthermore succeeded in detecting 24% of so far unknown PV panels in the cities of Bonn and Düren, which underlines the need for alternative ways of detecting PV panels for national registries. Nevertheless, improvements are required concerning the number of false-positive detections. Also, performance drops were detected caused by the distance between the training and validation area in NRW. This exposed the network to different geographic regions, with different urban planning and architectures, causing overfitting to one specific region. In contrast, stable results were achieved for training the model on the region of Heerlen and validating it on an area nearby. Furthermore, it was concluded that the difference between aerial images of 10 or 20 cm resolution lies in the ability to distinguish smaller objects, such as rooftop skylights.

2.7.2. Convolutional Neural Network for Semantic Segmentation

In a project conducted by Malof et al. (2017), previous shortcomings faced with traditional machine learning algorithms were tackled by shifting to CNNs, which achieved major improvements in object recognition. The applied CNN architecture was inspired by the designs of the Visual Geometry Group (VGG) at Oxford University (Simonyan and Zisserman, 2014). Overall, the model was trained based on circa 2.5 million training patches (aerial images) with a resolution of 30 cm, which were grouped into batches of 64 image patches to be trained for 16 epochs. The learning rate for the SGD was 0.001. Following their previous work, they closed the performance gap between training and testing data sets and achieved a recall rate of 80% and a precision of circa 95% on a testing data set. This project proves how CNNs are superior to traditional ML algorithms. Furthermore, the project examined the impact of utilizing transfer-learning (based on ImageNet), meaning that pre-trained weights were incorporated into a different model, so that the training of the model did not need to start from scratch. In contradiction to the results of Ševo and Avramović (2016), it turned

out that the use of pre-trained weight was not beneficial for the approach of Malof et al. (2017).

In a similar approach for mapping the location and size of PV panels, Castello et al. (2019) proposed a CNN with U-Net architecture for image segmentation of high-resolution aerial images. As in Malof et al. (2017)'s project, it outputs a semantic segmentation containing either PV class or no-PV class pixels. An important outcome of this work is the trade-off between solely including images with PV panels and adding images without PV panels. By adding images without PV panels, the model can learn various objects in the surroundings of PV panels, which might reduce the FP rate. On the other hand, having a relatively high percentage of target class pixels per image improved the precision of the model. This can be achieved by either considering smaller image patches or by including solely images that include pixels associated with PV panels. Overall, the proposed algorithm achieved an accuracy of 94%, an F1-score of 80%, and an IoU of 64% by utilizing 4680 images, grouped into batches of 32 images that are propagated through the U-Net for a fixed number of 75 epochs. The weights are adjusted by a the Adam optimizer by using a weighted pixel-wise categorical cross entropy function as loss function and a learning rate of 0.1 (Castello et al., 2019).

One of the most recent studies on semantic segmentation for detecting PV systems was published by Da Costa et al. (2021). Unlike previous studies, the focus was on solar plants and not on small-scale PV panels on rooftops. Also, Sentinel-2 imagery was used instead of high resolution aerial images as well as a NIR band in addition to the RGB bands. However, the effect of the NIR band is not examined. The project discusses the performance differences between four CNN architectures, namely U-Net, DeepLabv3+, Pyramid Scene Parsing Network, and Feature Pyramid Network, combined with four different backbones (Efficient-net-b0, Efficient-net-b7, ResNet-50, and ResNet101). In total, 290 images were used for training, validation, and testing purposes, which were grouped into batches of 5 images and propagated through the model for a fixed number of 300 epochs. The input patches were generated by applying a mosaicking technique that employs a sliding window to extract overlapping patches from one greater scene (Carvalho et al., 2021). This approach aims to eliminate classification errors at the edges of image patches. To adjust the weight the Adam optimizer was employed with a learning rate of 0.001 and the Dice Loss as loss function. Although the U-Net-Eff-b7 combination achieved the best results (Accuracy: 98.08%; IoU: 91.17%;F1-score: 95.38%), it needs to be highlighted that the performances of all model–backbone combinations were sufficient. The main outcome of their work is that results depend rather on the quality of the input labels than on the type of CNN. That is why they recommend following rather data-driven approaches to detect PV panels than model-driven approaches. This conclusion is backed by the fact that they obtain relatively poor results for PV panels located in residential buildings due to smaller PV system sizes compared to the solar plants, which they focus on (Da Costa et al., 2021).

2.7.3. PV panel types

To provide a better understanding of a PV panel's visual appearance, Figure 2.9 shows a generalized overview of PV panel types. In total, there are three types consisting of different materials that impact the efficiency rate of the panel (SolarReviews, 2021). The panels either distinguish themselves by color (black or blue) or by the appearance of the frame and grid that assembles the PV cells. While the color might be of great importance for analyses concerning the PV panel's environment, the silver frame and grid could be of great significance

for its detectability by CNNs. The latter is based on the fact that CNNs recognize patterns in images, such as the bright edges of the panel's metal frame. A potential challenge might be the differentiation between PV systems and solar thermal systems. Although STC are slightly larger than PV panels, both look similar in aerial images.

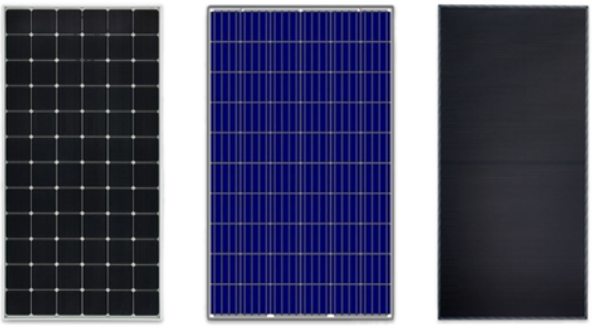


Figure 2.9.: (left) Monocrystalline, (middle) Polycrystalline, (right) Thin film (SolarReviews, 2021)

3. Methodology

The working steps carried out for this thesis are summarized in a workflow diagram intended to provide an overview of this thesis and allow its reproducibility. The purpose of each step in the workflow is explained in this chapter, while the technical details on how to implement them, are further discussed in Chapter 4.

As outlined in Figure Figure 3.1, the workflow starts with the definition of a study area for which aerial imagery, labeled data, and additional data need to be provided. Having the data, multiple pre-processing steps are required to prepare the data in such a way that it can be used by the CNN. The CNN with U-Net architecture is employed for training the detection of PV panels, and for classifying PV panels. In the final section of the workflow, evaluation methods are presented to assess the performance of the CNN.

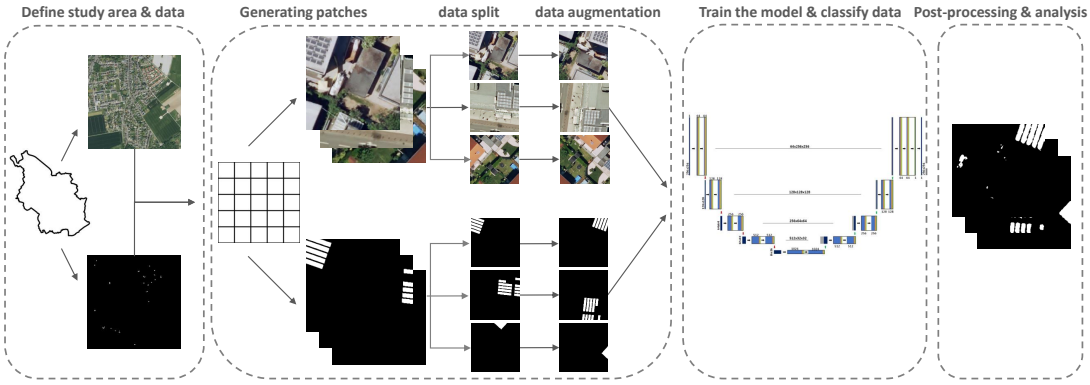


Figure 3.1.: Overview of workflow

3.1. Study area requirements

The main requirement for the study area is the availability of open spatial data. High-resolution aerial images from which ground truth data can be derived are the foundation for the classification and analysis in this thesis. Also, the availability of NIR data is essential for the second research question.

Furthermore, the study area requires certain properties that are suitable for answering the research questions. This is especially the case for the first and second research questions, since both are highly linked to urban characteristics, referring to land use types and rooftop colors. Against this backdrop, potential study areas need to be examined for a variety of both aspects. Based on the assumption that different land use types represent different building types and, therefore a variety in rooftop colors, it seems appropriate to divide the study area into simple differentiable land use types. Hence, the study area is divided into three

subareas, namely into a commercial area, a densely built-up area (city center) consisting of residential and mixed land use, and suburbs solely used for residential purposes. The selection of each subarea and respective analyses of its urban characteristics are presented in Section 4.1.4.

3.2. Ground truth data collection

The next step, after having each AOI defined, is the annotation process of PV panels. In this process, high diligence was required since manually drawn annotations shall represent the ground truth on which the CNN is trained. In large-scale projects, for instance, conducted by Bradbury et al. (2016), more manpower allows having multiple annotators labeling each image independently to ensure a high quality of ground truth data. As this is not the case for this thesis, each PV panel that was not identified as such with high confidence was not considered. Particularly, this applies to STC, which are challenging to distinguish from PV panels on aerial images due to their similar rectangular form and color. An indicator to differentiate them is the coverage of panels on rooftops since PV panel installations are usually more complex than compact STC (see Figure 3.2).

Figure 3.2.: Visual difference between PV panel (yellow polygon) and solar thermal collectors in an aerial image (left) and 3D imagery from Google Earth (right)

The annotations are drawn as polygons outlining entire PV panel systems. To improve the efficiency of finding PV panels in the image, repetitive checking of the same areas needs to be avoided. For this reason, a grid was used to systematically scan the areas. By marking each grid cell as *PV panel*, *no PV panel*, or *unsure*, areas were only checked once (see Figure 3.3).

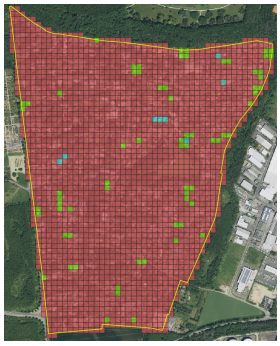
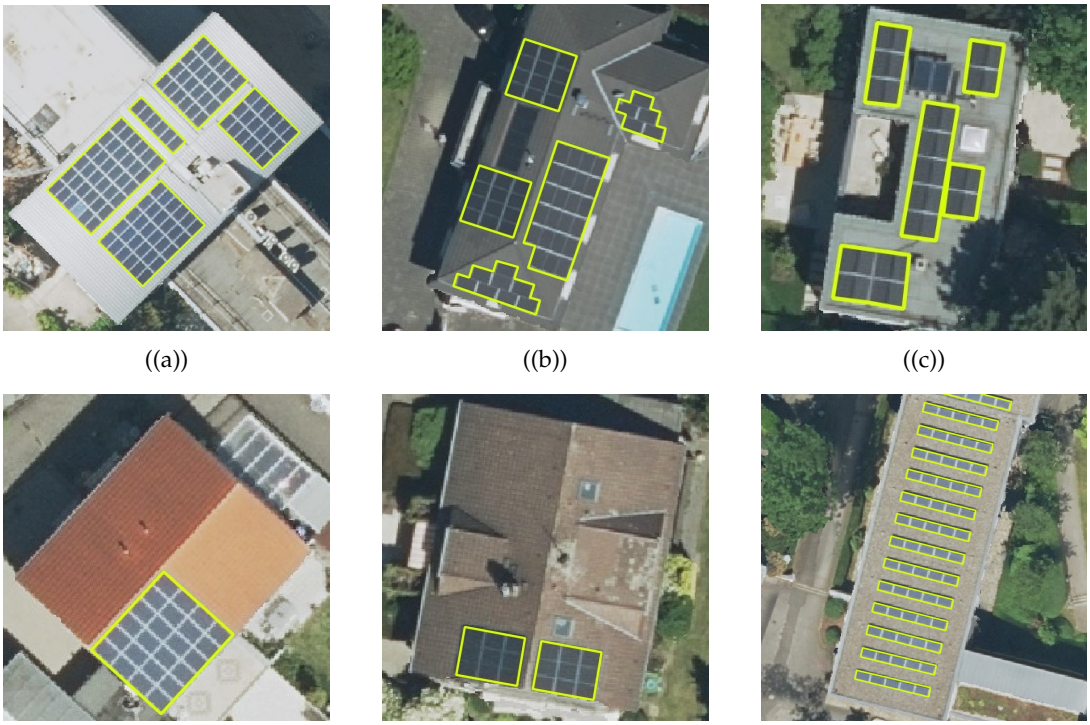


Figure 3.3.: Grid to structure the annotation process; *PV panel* (green), *no PV panel* (red), *unsure* (blue)

In the process of collecting ground truth labels additional properties were gathered to provide information about the actual differences between the subareas in Section 3.1. First, all PV panel arrays that belong together received a unique *system ID* to allow queries and statements on individual PV systems. Further, the number of PV panels per PV system was counted to provide information about PV system sizes. Additionally, rooftop colors were identified to determine predominate colors of each subarea, allowing for the analysis of the second research question. The variation of rooftop colors is presented in Figure 3.4.



((a)) ((b)) ((c))

((d)) ((e)) ((f))

Figure 3.4.: Rooftop color comparison: (a) white, (b) black, (c) greyish, (d) reddish, (e) brownish, (f) beige (sandy ground); PV systems outlined in yellow

Furthermore, the contrast of the rooftop color is set into context by considering the color of the PV panel. Therefore, the color of each PV system is documented.

3.3. Pre-processing of aerial images and ground truth data

In the next stage of the workflow (see Figure 3.1), aerial images and ground truth data (converted to a binary image) covering the same area must be processed in the same manner to serve as a model input. The technical processing of both aerial images and ground truth images is explained in more detail in Section 4.3.

3.3.1. Generating patches from orthophotos and ground truth labels

At this point of the workflow, it is important to consider the U-Net architecture structure that dictates the dimensions of the input data. Since it only takes fixed input dimensions (see Section 2.4) that need to be divisible by 2 to allow halving the dimensions with each max-pooling, it is required to generate uniform tiles for both inputs. Furthermore, the coverage of each tile needs to have an appropriate size to detect PV panels. A common input dimension for PV panel detection is 256x256 pixels (Castello et al., 2019; Da Costa et al., 2021). However, it is required to determine the dimensions with regard to the image resolution in meters. For instance, Castello et al. (2019) implement a convolutional layer of 256x256 pixels having a spatial resolution of 25 cm while Da Costa et al. (2021) apply the same dimensions for satellite images at a resolution of 10 m. Nevertheless, both dimensions are appropriate concerning the PV system size since Da Costa et al. (2021) aim to detect much larger PV system plants.

In this analysis, the input data has a spatial resolution of 10 cm. To find the appropriate patch dimensions, the footprint sizes of buildings (with PV system) incorporated in the classification are analyzed. Those footprint sizes range from 6 to 5,413 m^2 with a mean of 410 m^2. Consequently, having objects of around 20 x 20 m, an input dimension of 256x256 pixels (25.6 x 25.6 m) seems appropriate to capture building rooftops. For that reason, all input data are tiled image patches of 256x256 pixels.

Furthermore, it is significant to create an appropriate selection of image patches that is forwarded to the model. As Castello et al. (2019) indicated, having a relatively high percentage of pixels associated with PV panels per patch is beneficial to improve the precision. Against this backdrop and given that only little training data is available and the presence of relatively small PV systems in the suburbs, it might be beneficial to include solely image patches containing PV panel pixels.

3.3.2. Data split

The next step is the data split in training, validation, and testing data as described in Section 2.2.1. Common split ratio used in research for PV panel detection implies a suitable ratio of 70% for training data, 20% for validation data, and 10% for testing data (Castello et al., 2019; Da Costa et al., 2021; Kingma and Ba, 2014). The split into the respective data sets is conducted randomly to prevent a biased distribution of data.

3.3.3. Data augmentation

The final step before feeding the network with data is the augmentation of data. Applying this method is especially important for this thesis since it enriches the variety of input patches for the model. The augmentation is based on horizontal and vertical flips of input patches. This method enhances the robustness of the model by reducing the effect of overfitting, as proven by De Jong et al. (2020). Splitting needs to be carried out before data augmentation to avoid the case that augmented patches are part of more than one data set. Having the same patch in the training and test data set would not allow an unbiased evaluation of the model's accuracy since the patch was already seen by the model in the training data set

Typical data augmentation techniques, such as random changes in brightness, contrast, saturation, or hue, are not applied to the data to allow conclusions on the effect of unaltered roof colors.

3.4. Classification with U-Net

As explained in Section 1.2, the CNN architecture employed in this thesis is the U-Net by Ronneberger et al. (2015) presented in Section 2.4.

First, each augmented data set (training, testing, and validation) is grouped into batches that allow memory efficient adjustments of weights in terms of the mini-batch method described in Section 2.2.5. Secondly, the U-Net trains the detection of PV panels based on the training batches for a fixed number of epochs (or iterations). The validation batches provide unbiased insights into training progress. Lastly, the model is trained, meaning that the weights are adjusted in such a manner that the model should be able to detect PV panels in the testing data set. The final classification outputted by the model is a binary mask prediction of PV panels (semantic segmentation). Using these masks, quantitative and qualitative (refers to visual analysis) assessments can be conducted as described in the following Section 3.5.

The modification of the original U-Net architecture and the determination of its hyperparameters are described later in Section 4.4.

3.5. Evaluation of results

The assessment of the network's performance is divided into two approaches. First, the classification metrics described in Section 2.6 are applied to the predictions, followed by an analysis of the aerial image's MR within the region of the rooftop.

3.5.1. Classification metrics and visual assessment

The assessment of the model is based on the classification results of the testing data (10%). The metrics (accuracy, precision, recall, F1-score, and IoU) evaluate the overall performance of the model by calculating validation scores based on TP, TN, FP, and FN according to their formulas. Further, the results can be reasoned by visual comparisons between images,

labels, predicted probabilities (ranging between 0 and 1), and predicted masks (or prediction binaries). This comparison is, in particular, useful to visually reason the differences between precision and recall scores since they can be easily distinguished by considering FP and FN in the classification.

3.5.2. Mean reflectance analysis

Calculating the MR is a method that computes the mean of pixel values of a certain region in an image, for instance, the region can outline PV panels or rooftops to manually identify differences in reflectance. This analysis is of particular interest for answering the second and third research questions. It allows for distinguishing the MR values of multiple image channels (or bands). Thus, the impact of the NIR band can be examined more closely by comparing its MR value to those of RGB channels.

In total, the MR is calculated for five regions to analyze the classification from different perspectives. First, the PV panel prediction is considered as one region that is compared with the MR of the surrounding rooftop. By doing this, significant differences or similarities can be identified between both prediction and rooftop. The second analysis compares FP and FN located on a rooftop. In other words, the differences in MR between false PV panel predictions on the rooftop and the missing classification of a PV panel are analyzed. Lastly, the MR of the ground truth label is provided as a reference for the other regions. The results of this method are presented in Section 5.3.3.

4. Technical Implementation

This chapter describes in detail the technical implementation of the methodology presented in Chapter 3. First, the software and hardware are listed and reasoned, followed by an introduction to the study area of this thesis, after which each data set used is described in Section 4.2. Finally, technical details on pre-processing the data, implementing the CNN, and evaluating its performance are described.

4.1. Software and hardware used

4.1.1. QGIS

QGIS (version: 3.18.2-Zürich) is an open-source geographic information system (geographical information system (GIS)) that was used in this study for defining the study area (Section 4.1.4), generating ground truth labels (Section 4.2.2), and processing data (Section 4.3.1). Within QGIS, the geospatial data abstraction library (GDAL) (version: 3.1.4) was used as a plugin for raster operations such as GDAL Translate and GDAL Merge.

4.1.2. Software

Further, software used for implementing the methodology is the integrated development environment (IDE) RStudio (version: 2022.07.0+548) to utilize the R programming language (version: 4.1.2) for tiling aerial images and ground truth labels into patches (Section 4.3.2). The R packages *raster*, *sf*, *scales*, and *png* are used to process rasters, vectors, and arrays.

The Python programming language (version: 3.7.3 64-bit) was used with the Visual Studio Code (version: 1.72.2) IDE for splitting the input data (Section 4.3.3), generating a custom data set (Section 4.3.4), post-processing the output data (Section 4.6), generating a heat map (Figure 5.8), and in particular for constructing the CNN described in Section 4.4. The most important library for this thesis was the TensorFlow library (version: 2.8.2). TensorFlow is an open-source library for ML applications with particular attention on ANNs. Noteworthy libraries used in the post-processing are rasterio and scikit-learn.

The post-processing step of calculating the MR as described in Section 3.5.2 was carried out in the web application Google Earth Engine (GEE) using the programming language JavaScript.

4.1.3. Hardware

Instead of using the hardware of a local machine for running the model, the Colab (Colaboratory) from Google Research was used. It provides an online hosted Jupyter notebook service that allows access to the computing resources of GPUs.

4.1.4. Define study area

The study area of this thesis is located in the city of Cologne in the state of NRW in Germany. It fulfills the requirement of available open spatial data in terms of aerial images and additional vector data sets (see Section 4.2). As indicated in Section 3.1, the study area must be divided into three subareas to answer the research questions (see Figure 4.1). The northern subarea (1) is a commercial area in the district of Ossendorf in Cologne. The second district is the city center of Cologne which predominately consists of residential and mixed land use. Lastly, the residential areas of the southern districts of Hahnwald (3.1) and Meschenich (3.2) are chosen to represent the suburbs.

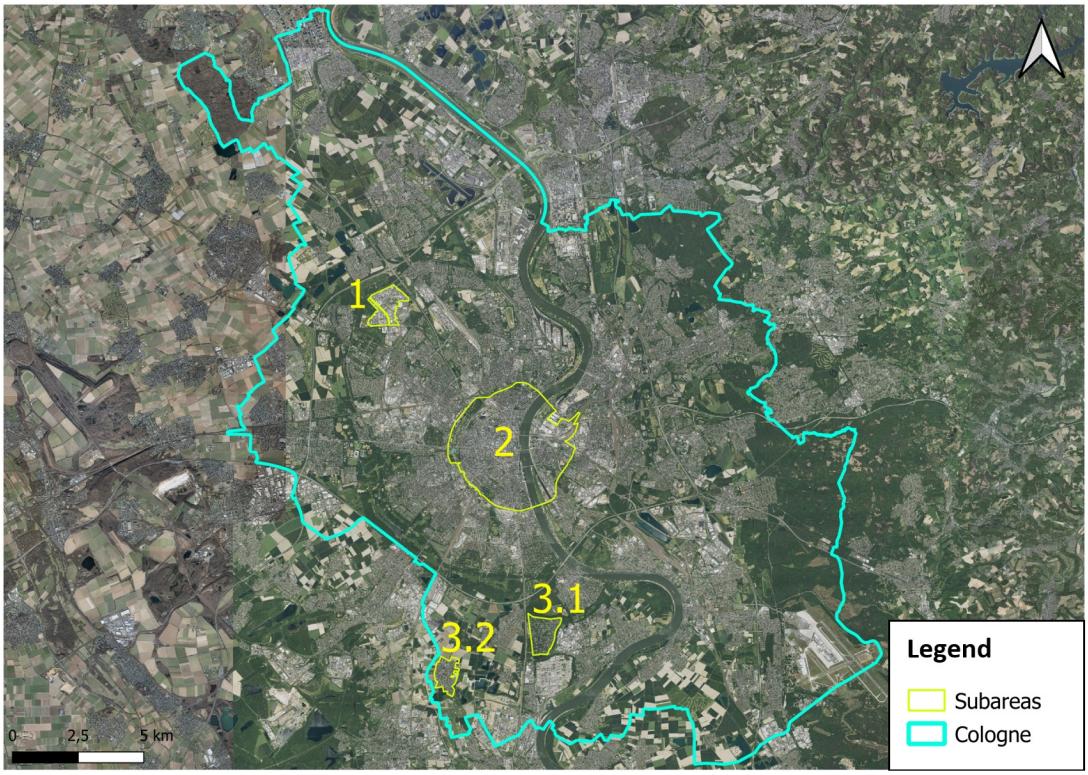


Figure 4.1.: Study area overview: (1) commercial area, (2) city center, (3.1) suburb (Hahnwald), (3.2) suburb (Meschenich)

Differences in the urban area are also proven by Table 4.1 showing a notably low building density in the commercial area, higher building densities in the suburbs, and the highest

density of buildings in the city center[1].

	commercial	city center	suburbs	**total**
Area (km^2)	1,393	7,694	2,227	**11,314**
Buildings	638	20,998	5,055	**26,691**
Buildings/km^2	456	2,729	2,270	**2,359**

Table 4.1.: Overview of building density in all areas (calculated based on building footprints per area)

4.2. Data sets used

This section describes all data sets utilized in this study. The key data sets are the aerial images and the manually generated ground truth labels. Additionally, a land use map and building footprints were incorporated and covered in Section 4.2.3.

4.2.1. True digital orthophotos

The aerial images are provided by the open spatial data infrastructure (SDI) Geobasis NRW as tiles with dimensions of 1 x 1 km in the format of JPG2000. Each tile has a resolution (ground sample distance (GSD)) of 10 cm and an average position accuracy of 2 to 3 pixels (20 - 30 cm). Furthermore, they consist of four spectral channels, namely RGB and NIR, with a radiometric resolution of 8 bits and a temporal resolution of 2 years. The images underlie the projected coordinate system ETRS89/UTM32 (EPSG 25832).

The aerial images were processed to distortion-free and true to scale images called digital orthophotos (DOPs). In an additional step, DOPs were rectified to TrueDOPs by adjusting tilting objects, e.g., buildings (see Figure 4.2) (GeobasisNRW, 2022). Therefore, TrueDOPs allow a vertical view on the image by eliminating blind spots[2] while preserving the geometric and radiometric qualities of DOPs (see Figure 4.3). This is also an advantage over other DOP providers, such as PDOK in the Netherlands (PDOK, 2022).

[1] In the following, the term *city center* refers to the areas of residential and mixed-use areas in the district of the city center without considering waterways, parks, etc.
[2] refers to not visible areas in the image due to buildings appearing tilted in the image

Figure 4.2.: Comparison between DOP (left) and TrueDOPs (right) (GeobasisNRW, 2020)

((a)) ((b)) ((c))

Figure 4.3.: (a) DOP with blind spot, (b) photo capturing with airborne camera, (c) TrueDOPs without blind spot (GeobasisNRW, 2020)

4.2.2. Ground truth labels

Below, a comprehensive overview of ground truth labels and their characteristics is given. In total, the data set contains around 12,508 PV panels spread over 171 buildings (manually counted). Table 4.2 indicates that the city center is representing the most average area of all three subareas in terms of PV panels per building and the mean building size. In general, in the commercial area, larger rooftops allow significantly larger PV systems than smaller rooftops in the suburbs. Figure 4.4 gives an impression of the visual extent of average PV systems. The position accuracy of the manually drawn ground truth labels is relatively high due to the basis of TrueDOPs for which the height of a building is not distorting the location of a PV panel.

	commercial	city center	suburbs	**total**
Buildings with PV panels	31	62	78	**171**
PV panels	7,994	2,431	2,083	**12,508**
Mean PV panels/building	258	39	26	**73**
Buildings (with PV panel) mean size (m^2)	1,364	418	140	**410**

Table 4.2.: Overview of PV panels in all areas

((a)) ((b)) ((c))

Figure 4.4.: Average PV system sizes (Number of PV panels): (a) commercial (295), (b) city center (40), (c) suburbs (28)

A strong variation in PV system sizes affects a balanced presence of target class pixels in the label patches. Table 4.3 documents the imbalance between target class pixels and the background pixel within a subarea based on all label patches of 256x256 pixels each. Furthermore, it shows a divergence amongst all subareas, particularly between the commercial area and the other two areas.

	commercial (%)	city center (%)	suburbs (%)
10 cm	19.16	5.38	3.98
20 cm	10.03	1.75	1.4

Table 4.3.: Average percentage of pixels associated with PV panels per label patch for each area at 10 and 20 cm resolutions

Further characteristics regarding the rooftop color are summarized in Figure 4.5, from which the predominant colors per subarea can be derived. According to the histogram, white is the most frequent roof color in the commercial area, and grey is the most frequent color in the city center. In contrast, no predominant roof color can be determined for the suburbs as the distribution of grey, black, and red rooftops is very similar.

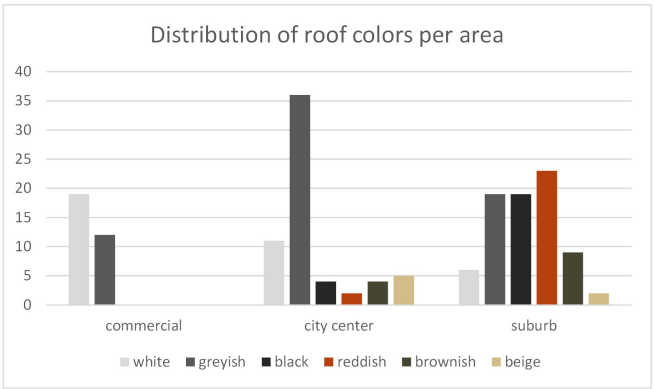


Figure 4.5.: Distribution of roof colors (of rooftops with PV panel) per area

The rooftop color might have different effects on the detection of PV panels depending on the PV panel color. In the following Figure 4.6, the percentage of blueish or black PV panels is summarized. It needs to be mentioned that blueish PV panels tend to appear in light grey in TrueDOPs when exposed in directed orientation towards the sun as well as dark blueish when they are opposed to the sun. Reflections can vary depending on the horizontal and vertical angles between the sun, the PV panel, and the airborne camera capturing the images. However, the comparison shows that blue is the predominant color of PV systems in all subareas. Nevertheless, around a third of the PV systems in the suburbs is black as well as a quarter of the PV systems in the commercial area. Another pattern observed in the images is the fact that in the commercial area all buildings with PV panels have flat roofs while buildings in the city center and the suburbs have flat or pitched roofs.

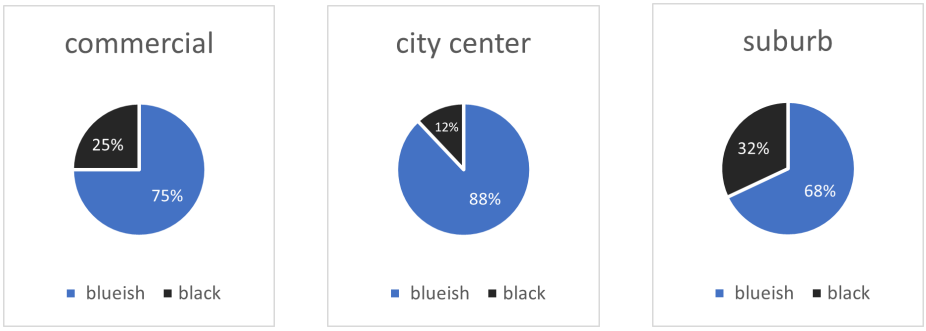


Figure 4.6.: Comparison of the PV panel colors per area (manually defined)

4.2.3. Additional data

Additional data sets incorporated in the methodology are building footprints and a land use map. The building footprints are derived from the authoritative real estate cadastre information system of NRW and provided in a vector format (OpenNRW, 2022b). The land use map shows the existing and planned land use of the entire city area, including the three study areas (OpenNRW, 2022a).

4.3. Pre-processing steps

The technical implementation of each working stage described in Section 3.3 is explained in the following.

4.3.1. Processing TrueDOPs and ground truth labels

First, the TrueDOPs are merged per subarea in QGIS. The merged output is converted from a JPEG2000 to a TIFF raster format due to processing issues that occurred with the JPEG2000 raster. To reduce the computational demand when working with large data sets, the merge of all subareas TrueDOPs is conducted by building a virtual layer. This layer intends to provide the spatial extent of all rasters for the rasterization of ground truth labels. The ground truth

labels are rasterized in QGIS, outputting a binary raster showing PV panels with a value of 1 and background information as *no data*.

This process was performed once at a resolution of 10 cm and once for downsampled data at a lower resolution of 20 cm.

4.3.2. Generating DOP and ground truth label patches

The same grid used for collecting ground truth data (see Section 3.2) is reused to generate input patches cropped to the dimensions of 256x256 pixels. Therefore, an additional grid needs to be generated for the resolution of 20 cm. Both grids are generated (in QGIS) consisting of georeferenced polygons with grid cells of the size 25.6 x 25.6 m (10 cm resolution) and with 51.2 x 51.2 m (20 cm resolution). To consider only patches that contain target class pixels, only those grid cells are extracted that intersect with the ground truth data.

The selection of grid cells is loaded in an R script to crop TrueDOPs and the ground truth binary raster once to patches in PNG format and once to TIFF patches. The PNG format is chosen due to the TensorFlow data set function that restricts the use of data formats other than BMG, GIF, JPEG, or PNG. Saving patches as TIFFs is required to retrieve spatial reference information in the post-processing (see Section 4.6).

Table 4.4 provides an overview of the number of patches on which the model is trained.

Resolution	commercial	city center	suburbs	**total**
Number of patches (10 cm)	100	100	100	**300**
Number of patches (20 cm)	50	77	73	**200**

Table 4.4.: Overview of patches per subarea that contain PV panels

4.3.3. Data split

A Python script is employed to read the folder of all patches using the Python library *random* for splitting the patches into 90% training and 10% testing patches. The training patches are split again at a later stage to obtain 70% training and 20% validation patches. The file names are written into two lists (as text files) according to the training and testing split.

4.3.4. Compiling data to custom data set for TensorFlow

The patches of both TrueDOPs and labels of all areas are uploaded to the same directory in separate folders on a Google Drive to be accessible for Google Colab. Moreover, both lists defining the data split are uploaded to the folder of label patches on Google Drive.

In the Jupyter Notebook hosted in Google Colab, a TensorFlow data set (TFDS) is initialized using the command-line tool TFDS command line interface (CLI). In this data set, a Python script defines the source of data, the format of the data, and the data split. It is required to modify the script so that it allows raster labels as input, and knows the respective input dimensions as well as the source path to the respective Google Drive folder. Then, the TFDS CLI is utilized to download and prepare the custom data set on-the-fly in Google Colab.

4.3.5. Additional preparation steps

After loading the data, the training patches are split into training and validation, followed by a normalization of each patch from unsigned 8 bits integers to decimal numbers between 0 and 1. Then, all patches are shuffled before being grouped into batches. As indicated by Da Costa et al. (2021), a batch size of 5 seems appropriate when dealing with around 300 images. In the final step before forwarding the batches to the model, the patches are augmented as described in Section 3.3.3.

4.4. Modified U-Net architecture and hyperparameter definition

The U-Net architecture employed in this thesis is a slightly modified version (see Figure 4.7) of the original architecture presented in Section 2.4. Its main differences lie in the additional batch normalization layer (see Section 2.5.1) between the convolutional layer and the ReLU activation (see Section 2.2.3) function. Furthermore, the dimensions of the feature maps do not decrease after convolutional operations due to implemented zero-padding operations (explained in Section 2.3.1). Therefore, the original *copy and crop* operations are replaced by a simple copy operation to reconstruct the same spatial resolution at each level of the U-Net. Additionally, the U-Net is designed to take patches of 256x256 pixels as input. The size of the x- and y-dimensions is discussed in Section 3.3.1. Further, the selected input dimensions meet the requirement of having tiles with even x- and y-dimensions which allow seamless tiling after each max-pooling operation with a kernel size of 2x2 (Ronneberger et al., 2015). The third dimension, which defines the number of image channels, can be manually changed from 3 to 4 channels to either process RGB or RGB plus NIR images.

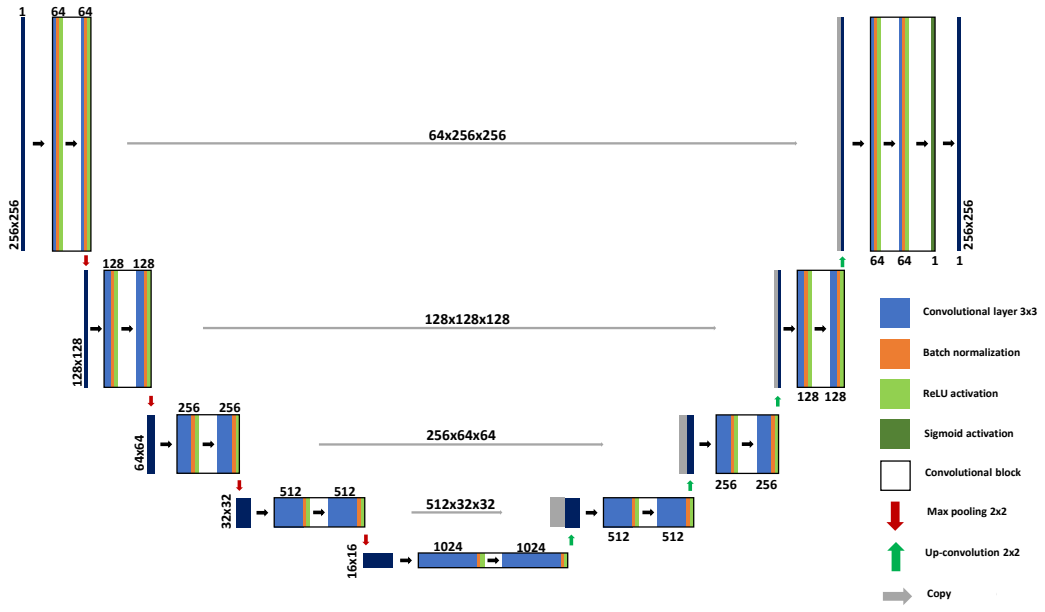


Figure 4.7.: Modified U-Net architecture; Dark blue boxes represent input or output features

Nevertheless, basic hyperparameters defined in the original U-Net architecture remain the same such as the ReLU activation function. The same applies to convolutional kernels keeping the size of 3x3 and a stride of 1, which was also utilized in similar studies (Castello et al., 2019; Malof et al., 2017). Similarly, the max-pooling operation with a kernel of 2x2 pixels and a stride of 2 is adapted. Further, the number of convolutional layers doubles with each max-pooling operation from 64 down to 1024 layers.

The weights are initialized with the He uniform variance scaling initializer described in Section 2.2.2, which is provided as a TensorFlow function. Transfer learning with pre-trained weights is not employed due to the NIR band which is denoted as the fourth image channel. Most pre-trained weights are based on RGB channels only. An overview of the total amount of trainable parameters is provided in the following Table 4.5. It is defined by the sum of weights and biases that can be adjusted during the training process.

Network	Trainable parameters
U-Net for RGB	31,043,521
U-Net for RGB+NIR	31,044,097

Table 4.5.: Number of trainable parameters

4.4.1. Loss function and optimizer used

The imbalance between target class pixels and background information presented in Table 4.3 results in uncertainty about the appropriate loss function to implement. As indicated in Section 2.2.4, loss functions differ to calculate appropriate losses for different use cases. Therefore, the uncertainty is faced by fine-tuning the learning process of the model with regard to the loss function. According to Section 2.2.4, both, the BCE (Equation 2.6) and the FL (Equation 2.7) seem suitable to be implemented in this study. A more precise comparison in Table 4.6 shows the performance results for all areas combined[3].

Area	loss function	accuracy (%)	precision (%)	recall (%)	F1-score (%)	IoU (%)
all areas	BCE	98.87	**95.32**	87.29	91.13	91.25
all areas	FL	**99.21**	94.36	**93.76**	**94.06**	**93.97**

Table 4.6.: Evaluation of U-Net with BCE using RGB TrueDOPs; Learning rate = 0.001; Number of epochs = 100

According to Table 4.6 the FL achieves slightly better classification results than the BCE. Nevertheless, the loss curve shows two minor spikes in the BCE (Figure 4.8) and one major spike in the FL curve (Figure 4.9), which significantly affect the learning progress measured by the F1-score.

[3]values in bold denote the better result

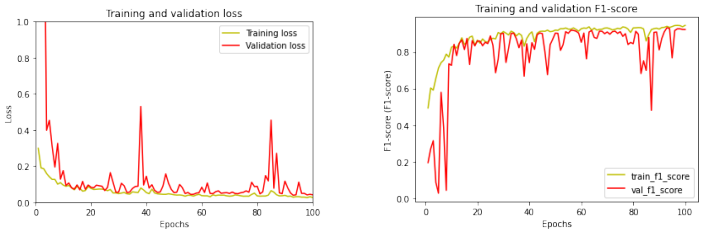


Figure 4.8.: Training and validation loss function and F1-score for each epoch (BCE)

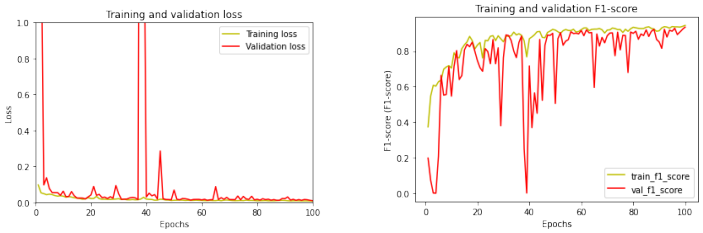
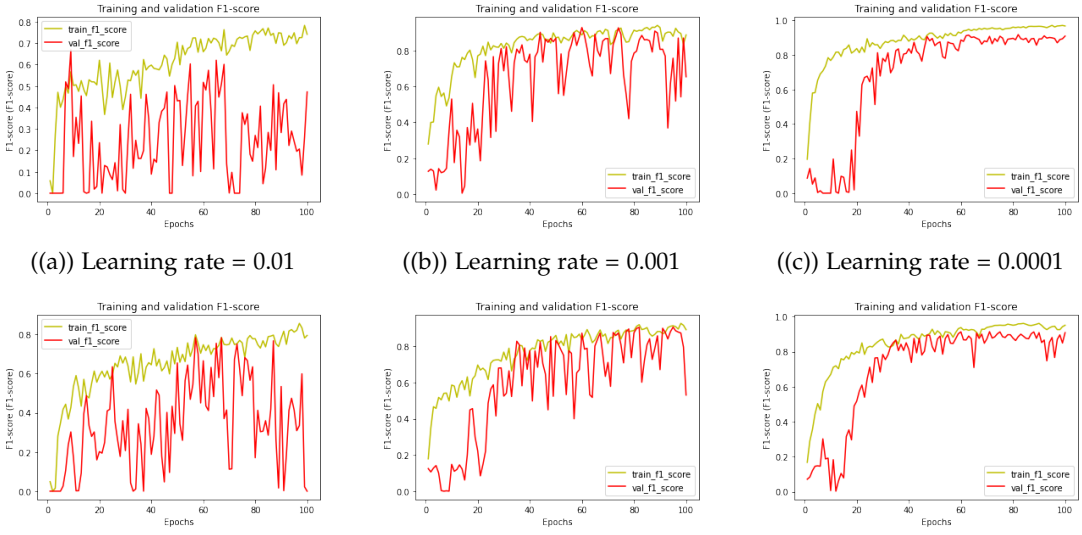


Figure 4.9.: Training and validation loss function and F1-score for each epoch (FL); (At epoch 40 the tip of the spike is at 4.5)

Considering smaller data sets of the subareas, both loss functions are compared again with attention on the learning rate to tune the learning process of the Adam optimizer. For each loss function, learning rates of 0.01, 0.001, and 0.0001 are implemented in the following.



((a)) Learning rate = 0.01 ((b)) Learning rate = 0.001 ((c)) Learning rate = 0.0001

((d)) Learning rate = 0.01 ((e)) Learning rate = 0.001 ((f)) Learning rate = 0.0001

Figure 4.10.: Model's performance according to F1-score. First row: BCE; Second row: FL

loss	LR	accu. (%)	prec. (%)	recall (%)	F1-score (%)	IoU (%)
BCE	1e-2	96.91	84.74	22.63	35.72	59.31
BCE	1e-3	98.11	90.49	55.93	69.13	75.45
BCE	1e-4	**99.23**	93.97	85.1	**89.31**	**89.95**
FL	1e-2	96.21	0	0	0	48.1
FL	1e-3	98.04	**98.39**	49.17	65.57	73.39
FL	1e-4	98.90	84.57	**86.76**	85.65	86.88

Table 4.7.: Evaluation of U-Net based on FL and BCE with different learning rates using RGB TrueDOPs of the city center; Epochs = 100

Figure 4.9 shows a similar behavior of both loss functions in the training process. Based on the results of Table 4.7 showing the best U-Net performance for the combination of the BCE loss function and a learning rate of 0.0001, both the loss function and the learning rate are selected for the experiments outlined in Section 4.5. Furthermore, it becomes evident that the accuracy metric is not suitable for assessing the performance of the model considering the disparities between precision and recall while having high accuracies.

To avoid over- or underfitting, the early stopping method (Section 2.5.2) is tested with a threshold (called patience) of 10, meaning that the training process stops when the validation loss has not improved for 10 epochs. In three consecutive training runs, the point of early stopping yields 38, 58, and 100 epochs. This variation is caused by fluctuations in the loss function. Therefore, it is required to force the model to train for a certain number of epochs to guarantee acceptable and comparable periods of training. Reading from Figure 4.10(c), the point at which the validation loss starts to stabilize while the training loss continues to increase is found around epoch 60. Based on this observation, a fixed number of 60 epochs is picked.

Due to the comparatively low number of 100 samples (for subareas) from which 70% are used for training, a batch size of 5 is chosen.

4.5. Training and testing experiments

After having the model defined, it is applied in multiple scenarios to serve the research questions. In total, three kinds of experiments are carried out. The first experiment analyzes the impact of training a model on different land use types. For this experiment, the U-Net is trained and tested on each subarea as well as on all areas at once. Further, the performance of each subarea model is evaluated based on cross-validations conducted on all other subareas respectively. While the term cross-validation commonly refers to the data split described in Section 3.3.2, this study employs the term to refer to the training and evaluation across multiple land use types. Further experiments are conducted to analyze the impact of including NIR data in the training process to meet the third research question. The last research question is examined considering input data of different resolutions. An overview of all scenarios is listed below:

1. Training and evaluating a U-Net within the same area based on TrueDOPs at a resolution of 10 cm with RGB channels:

- All areas combined (300 images).
- Each subarea on its own (100 images).

2. Evaluating the U-Net's performance based on cross-validation :
   - model trained on commercial area → evaluated on city center
   - model trained on commercial area → evaluated on suburbs
   - model trained on city center → evaluated on commercial area
   - model trained on city center → evaluated on suburbs
   - model trained on suburbs → evaluated on commercial area
   - model trained on suburbs → evaluated on city center
3. Evaluating the U-Net's performance by training and evaluating with NIR data:
   - All areas combined with NIR data
   - Each subarea with NIR data
4. Training and assessing the performance of the U-Net on lower resolution TrueDOPs:
   - All areas combined at 20 cm resolution
   - Each subarea at 20 cm resolution

4.6. Post-processing result for evaluation

As described in Section 3.5 there are two approaches to evaluate the model's performance. For the analysis of the classification metrics and the visual output, no further post-processing steps are required. To only consider the output on rooftops, spatial operations are applied in the second approach, requiring the data to be georeferenced. Since the input and output of the model are patches in PNG format, the original reference system needs to be reassigned to the patches. For that reason, the patches were saved as TIFFs in Section 4.3.2, so that the spatial information of the original TIFF can be linked to an output PNG. The georeferenced images, prediction masks, and labels serve as input in the following evaluations.

5. Results and analysis

Following Chapter 4, in which the technical implementation of the model and the corresponding pre- and post-processing steps are explained, this chapter presents and analyzes the performance of the U-Net regarding the key aspects of the research questions. It follows the structure of the experiments outlined in Section 4.5. Starting with the analysis of the classification of all areas as well as each subarea on its own. All areas are quantitatively evaluated based on classification metrics followed by visual analyses of sample results. Section 5.2 presents the quantitative results of the cross-validation between the areas. In Section 5.3 the impact of the NIR channel is described based on classification metrics, visual comparisons, and analyzes of the prediction's MR. Lastly, the classification performance of lower-resolution TrueDOPs is presented. It needs to be mentioned that all sample images shown in the following four sections are examples from different testing batches which were not seen by the U-Net before.

5.1. Classification of all areas based on RGB TrueDOPs

The results presented in this subchapter are computed by training the U-Net on RGB TrueDOPs of each subarea followed by a classification of the same area. The quantitative results in Table 5.1 and the visual outputs are entirely based on the testing data sets of each area which are not exposed to the U-Net during the training process. The same applies to the combination of all subareas. As summarized in Table 4.4, each subarea is covered by 100 patches, while the combination of all subareas consists of 300 patches. Accordingly, each test data set consists of around 10 patches (10%), whereas the test data set of all areas contains around 30 images.

5.1.1. Quantitative evaluation of RGB classifications

The U-Net's performance for each area is summarized by multiple classification scores in Table 5.1.

Area	precision (%)	recall (%)	F1-score (%)	IoU (%)
commercial	89.40	**91.5**	**90.44**	88.96
city center	89.1	85.59	87.31	88.25
suburbs	**97.86**	60.66	74.89	78.96
all areas	91.64	88.74	90.16	**90.36**

Table 5.1.: Classification results of each subarea and all areas combined

Considering the F1-score and the recall the best performance is achieved by the U-Net trained on patches of the commercial area. Both networks trained on commercial areas and all areas show the most consistent performance scores around 90% (±1.7%). The network based on city center patches achieved approximately high and constant results between 85 and 90%. Except for achieving the highest precision score, the network trained on suburb patches yields the poorest scores in the recall, F1-score, and IoU. Consequently, the network tends to predict PV panels only at those locations where PV panels are actually installed while it is prone to omit PV panels in the classification.

5.1.2. Visual evaluation of RGB classifications

In the following, examples of classifications of each area are presented. The examples are selected based on the presence of special features which allow detailed statements about the performance of the U-Net. Each example covers four images, the input image, the ground truth label (or true mask), the predicted probability, and the binary classification (or predicted mask). The predicted probability presents the direct output of the sigmoid activation function showing the probabilities between 0 and 1 of a pixel representing a PV panel (see Section 2.2.3). A threshold of 0.5 is splitting the probabilities into background pixels and target class predictions. The probability is included to observe pixels with low probabilities that might indicate potential misclassifications.

Commercial area. The first example in Figure 5.1 shows a successful classification on the rooftop while indicating a minor confusion by a staircase (leading to the building on the right) resulting in a few FP predictions. Nevertheless, it proves the capability of differentiating PV panels from the rectangular shape of a glass roof at the bottom of the image.

City center. The next example (Figure 5.2) reveals difficulties in the prediction of the darker PV panels at the lower right edge of the image. Further, it shows how shadows cause FN predictions which are represented by the missing detection of the right end of the PV panel array. Overall, the presence of shadows in the input samples is rare since most PV panels are installed at predominantly sunlit locations.

According to Figure 5.3, the network is having difficulties in detecting black PV panels in the city center. In the following paragraph, this type of PV panel is analyzed in a suburban setting (see Figure 5.5).

Suburbs. In both, Figure 5.4 and Figure 5.5, the U-Net is facing classifications of black PV panels in the suburbs. In the first image, the PV panel's appearance is in sharp contrast with the light grey rooftop facilitating the detection of almost all PV panels. Further, the PV panel edges consist of white frames which highlight the boundaries of the panel. Contrary to this, the PV system in the lower left of the next figure is representing a homogeneous black surface without bright edges, installed on a dark rooftop. The predicted probability of that patch indicates a little recognition of the PV system's outer boundaries whereas the inner surface is not considered at all. Consequently, the detection of the entire PV system is omitted.

Furthermore, FP predictions are caused due to the confusion of a PV system and the glass roof of a conservatory in Figure 5.5.

Similar to the misclassification of the conservatory, Figure 5.6 shows a FP prediction at the exact position of a skylight. Nevertheless, a close-by skylight at the left edge of the patch is classified correctly (as background pixels) since it appears less blueish in the image.

Lastly, the network proves the ability to deal with PV panels and STC in one image, as shown in Figure 5.7. Despite the similar blueish color and the rectangular shape, both STC are not classified as PV panels. Further observations have shown that STC are predominantly installed on rooftops of private households rather than on apartment buildings.

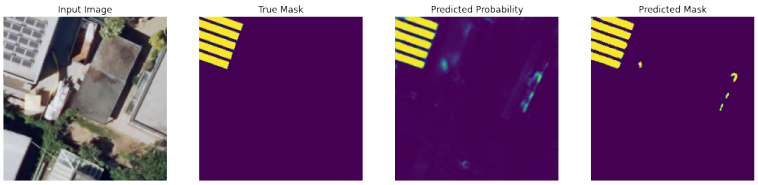


Figure 5.1.: Example located in the commercial area and based on the network trained on all areas; Special feature: **Glass roof**

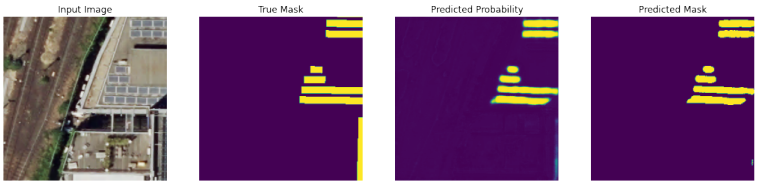


Figure 5.2.: Testing sample located in the city center; Prediction by network trained on the city center; Special feature: **Shadow**

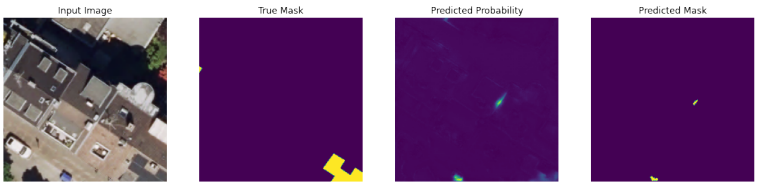


Figure 5.3.: Testing sample located in the city center; Prediction by network trained on the city center; Special feature: **Black PV panels**

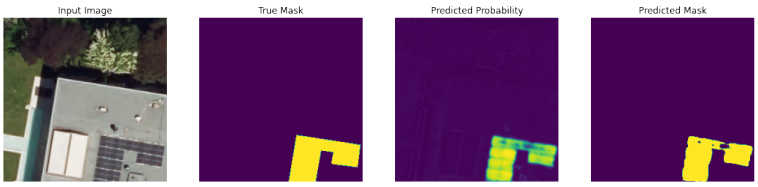


Figure 5.4.: Detection of black PV panels in suburbs

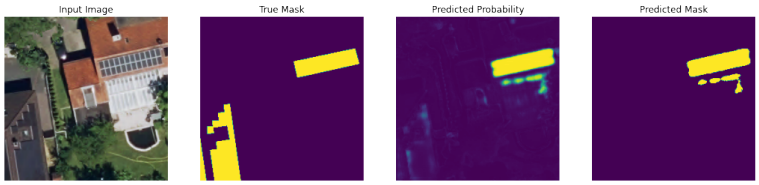


Figure 5.5.: Misclassification of black PV panels in the suburbs; Special feature: **Conservatory**

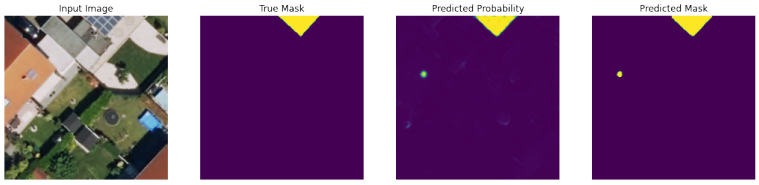


Figure 5.6.: Misclassification in the suburbs; Special feature: **Skylight**

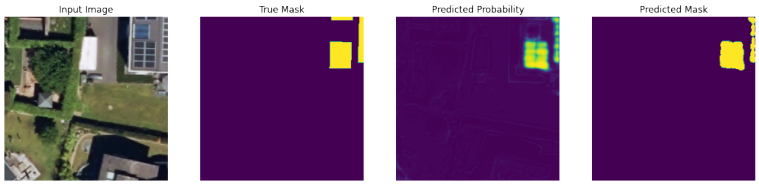


Figure 5.7.: Testing sample located in the suburbs; Special feature: **STC**

As observed in Figure 5.2, Figure 5.3, and Figure 5.5, the model has difficulties detecting black PV panels. It can be assumed that this type of PV panel (Figure 5.3 and Figure 5.5) without bright frames hinders the detection of patterns. Especially in combination with dark rooftops, the detection rate of the PV color might decline due to little contrast. However, the presented FN predictions are all located at the patches' edges. To analyze whether this could be an artifact (a systematic anomaly produced by the model), all FN predictions are overlapped within one patch (see Figure 5.8). This overlap creates a heat map of errors, so systematic errors can be identified by their location in the patch. The heat map shows one hotspot of errors in the center-left and one hotspot in the lower right corner of the patch. Overall, it proves that there is no artifact at the patches' edges, which reinforces the previous assumption that the model has difficulties in detecting black PV panels without bright frames, that are located on dark rooftops.

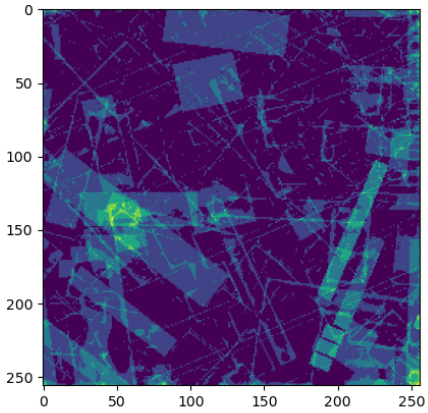


Figure 5.8.: Heat map of all FN predictions computed from 56 RGB testing images of all areas and each subarea, see Appendix B (from purple = no FN, to yellow = multiple FN)

Overall, the following three subfigures demonstrate how well the U-Net can train on each subarea. The network for the commercial area can converge to the F1-score of the training data after around 20 epochs. In contrast, the city center network is less consistent while the suburb network faces much more fluctuations as well as more epochs to converge.

It becomes evident that areas with a higher average of target pixels per patch achieve better F1-scores and start to stabilize at earlier stages in the training process (see Table 4.3). This is the result of larger PV system sizes in commercial areas, which provide uniform patterns of long panel arrays and homogeneous colors to learn by the U-Net.

((a)) commercial area ((b)) city center ((c)) suburbs

Figure 5.9.: U-Net's training and validation performance according to F1-score per subarea

5.2. Cross-validation: commercial area, city center, and suburb

Conducting cross-validations indicate which areas are most suitable for training a network that is capable of classifying other areas. It can also be considered as the most average area that comprises key features of different areas.

Table 5.2 summarizes the F1-scores of all cross-validations as well as the results of training and predicting networks based on images of the same subarea. It stands to reason that each network performs best on the area on which it was trained. The best results of the cross-validations across different land use types are highlighted in bold.

trained/predicted	commercial (%)	city center (%)	suburbs (%)
commercial	90.44	**72.89**	61.85
city center	**59.82**	87.31	**77.73**
suburbs	48.52	63.49	74.89

Table 5.2.: F1-scores of cross predictions

Overall, training images of the city center suit best for classifying commercial areas or suburbs. Vice versa, the network trained on images of the commercial area achieves the best classification of the city center. The poorest F1-score result of less than 50% is achieved by a network trained on suburb images classifying images of commercial areas. The results demonstrate the greatest discrepancies between commercial areas and suburbs while showing that a network trained on images of the city center serves as the best allrounder for classifying different land use areas. To provide a visual impression of the results, three samples on which the cross-validation is applied are attached in Appendix C.

The fact that the greatest discrepancy lies between the commercial area and the suburbs is also reflected by their differences in terms of predominant roof colors, variations of roof color, the number of PV panels installed, and the mean size of buildings with PV systems.

5.3. Classification based on TrueDOPs including NIR data

In this subchapter, the overall performance of a U-Net trained on RGB-NIR images is evaluated. Both training and evaluation are conducted based on the same testing patches used in the previous sections. In addition to the 3-channel RGB images, the NIR channel is included, resulting in four-channel images. First, the quantitative results per area are presented, followed by a performance comparison between RGB and RGB-NIR image classifications. Further, a qualitative analysis of this comparison is carried out by considering the visual differences between the new results and certain examples from the previous subchapter. Lastly, the impact of the NIR channel is analyzed in more detail based on its MR.

5.3.1. Quantitative evaluation of RGB+NIR classifications

Despite the high precision of U-Nets trained on images of the suburbs, the best classification performance is achieved by a network trained on images of all areas (see Table 5.3). Similar to the previous results (see Table 5.1) the suburb network obtains the greatest gap of around 44% between precision and recall scores.

Area	precision (%)	recall (%)	F1-score (%)	IoU (%)
commercial	93.91	84.07	88.72	87.35
city center	92.07	83.41	87.53	88.45
suburbs	**96.81**	52.65	68.21	74.71
all areas	94.06	**89.55**	**91.75**	**91.81**

Table 5.3.: Evaluation of RGB-NIR image classification

The comparison of classification scores between RGB and RGB-NIR-based networks (see Figure 5.10) shows a performance drop when the NIR channel is included in the classification of suburbs. In contrast, both, the commercial and the city center networks perform comparatively consistently in the classification of both image compositions. A minor performance drop can be identified in the classification of RGB-NIR images of the commercial area while the performance slightly increases in the case of the city center. Despite both negative trends, the performance of the network trained on images of all areas increases.

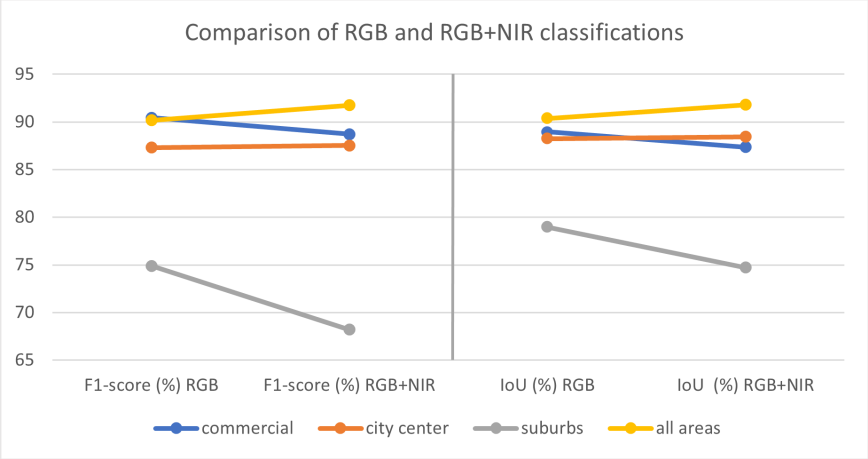


Figure 5.10.: Comparison of RGB and RGB-NIR-based classifications assessed with F1-score and IoU

5.3.2. Visual evaluation of RGB+NIR classifications

Commercial area. Figure 5.11 shows the improved classification result compared to the RGB image classification in Figure 5.1. This RGB-NIR sample proves the slightly increased performance of the network trained on images of all areas. This does not apply in the case of the patch showing a skylight (see Figure 5.6). Both classifications are nearly identical.

City center. A further improvement can be observed in Figure 5.12. While the RGB image-based classification (see Figure 5.3) is not able to detect the black PV panels in this patch, the additional NIR channel allows a partial classification. The comparison of both classifications indicates an improved capability of detecting bright rooftop edges as shown by the predicted probabilities in Figure 5.12. Although the edges are represented with little confidence, the probabilities appear clearer than in Figure 5.3.

Suburbs. The drawback of detecting bright edges more easily becomes evident in Figure 5.13. Although lower probabilities are assigned to the terrace than to the PV panels on the rooftop, the probabilities are high enough to exceed the threshold to generate a binary mask. Nevertheless, the prediction of the black PV panels improved since the coverage of the predicted mask is more coherent than in Figure 5.4.

Despite the improvements in classifying black PV panels, the black PV panels in Figure 5.14 are still not detected by the network. As explained in Section 5.1, it is assumed that the frameless type of PV panel and the little contrast to the rooftop cause the misclassification. Another aspect that might reinforce the misclassification is the case of having two different types of PV panels in one image patch. In that case, the U-Net must output similar prediction probabilities for two objects that belong to the same class but differ visually.

Furthermore, the classification of Figure 5.7 worsens when a NIR channel is added to the RGB image as shown in Figure 5.15. The network confuses the STC with PV panels as well as with a black rooftop in the lower left corner of the image patch.

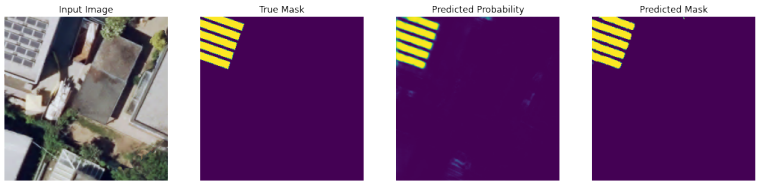


Figure 5.11.: RGB-NIR image classification in commercial area and based on the network trained on all areas; Special feature: **Glass roof**

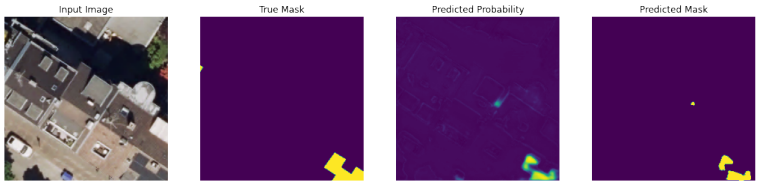


Figure 5.12.: RGB-NIR image classification of the city center; Special feature: **Black PV panels**

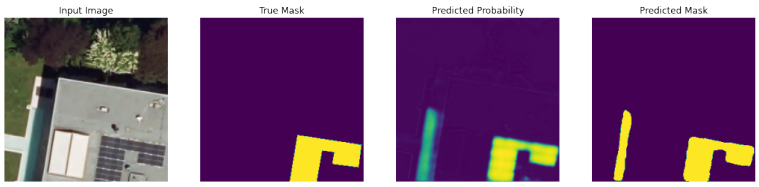


Figure 5.13.: Misclassification of terrace; Special feature: **Black PV panels**

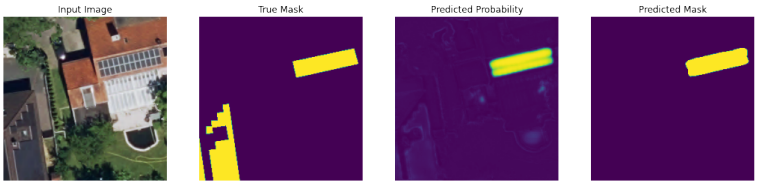


Figure 5.14.: RGB-NIR: Misclassification of black PV panels in the suburbs; Special feature: **Conservatory**

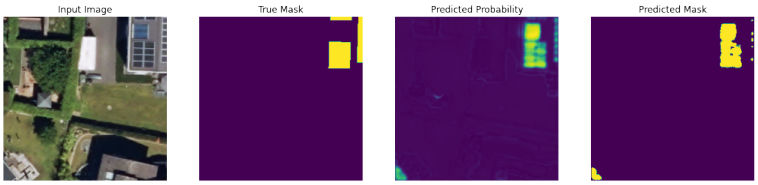


Figure 5.15.: RGB-NIR: Misclassification of STC in the suburbs

5.3.3. Analysis of mean reflectance

This section provides a closer look at the impact of a NIR channel by comparing the MR of PV panel predictions, ground truth labels, FP, FN, and TN. The MR values are solely based on pixels within a building's footprint. It intends to differentiate the prediction errors in a more detailed approach as well as to provide an impression of how a NIR might contribute to those errors or even improves the classification. Therefore, some results are also compared to the classifications presented in Section 5.1.

City center. Figure 5.16 shows the MR curves of the corresponding regions in the left image, represented in the respective colors. The MR indicates a significant mismatch between the ground truth label and the FN predictions that causes misclassifications of blueish PV panels in the shadow and black PV panels on the right edge of the image. Further, FP are located at the boundary of PV panels causing a similar MR as the PV panel prediction. The MR of the NIR channel is for most regions on a similar reflectance level as the RGB channels.

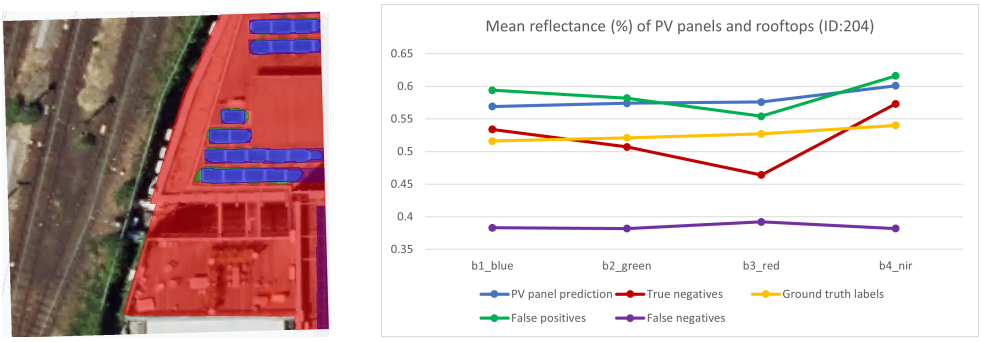


Figure 5.16.: MR of RGB-NIR testing sample located in the city center; Prediction by network trained on the city center; Special feature: **Shadow**; the colors in the diagram correspond to the polygon colors in the image (ground truth labels are not visualized)

An exception of the flat reflectance curve is the red region representing TN predictions. This curve indicates a much lower reflectance of the red channel than the ground truth reflectance as well as a higher reflectance of the NIR channel. The curve's shape is therefore slightly reflected by the FP curve. Nevertheless, the notable increase of MR between the red and the NIR channels might be caused by the balcony plant reflecting NIR radiation much stronger.

The next two figures compare the results of RGB and RGB-NIR-based classifications (see Figure 5.17 and Figure 5.18). In all regions, the mean NIR reflectance is higher than the respective reflectances of the RGB channels. Despite the discrepancy between the MR of TN and FN predictions, the PV panels are partially classified as background pixels. Furthermore, both classifications contain FP predictions at the same spot in the center of the image. Since the spot only covers a fraction of a homogeneous rooftop in terms of colors, it can be assumed that not only the reflectance is determining its classification. As CNNs can learn patterns, it is likely that the prediction is based on the combination of the dark roof color (of the rooftop side orientated to the northwest) and the pattern of parallel bright edges of a dormer and the boundary to the next rooftop. This pattern would correspond to the bright (e.g., silver or white) frame that typically bounds PV panels. While the same pattern can also be found on the other side of the rooftop, it seems to have a different reflectance due to the sunny side of the rooftop being orientated to the southeast.

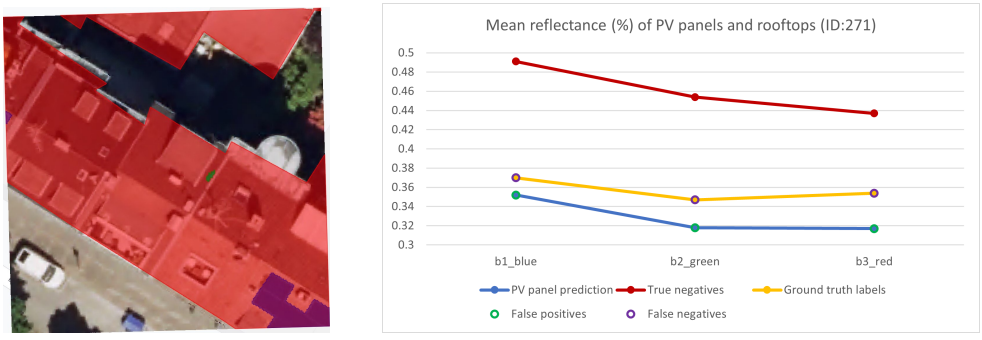


Figure 5.17.: RGB: Classification example of the city center; Special feature: **Black PV panels**

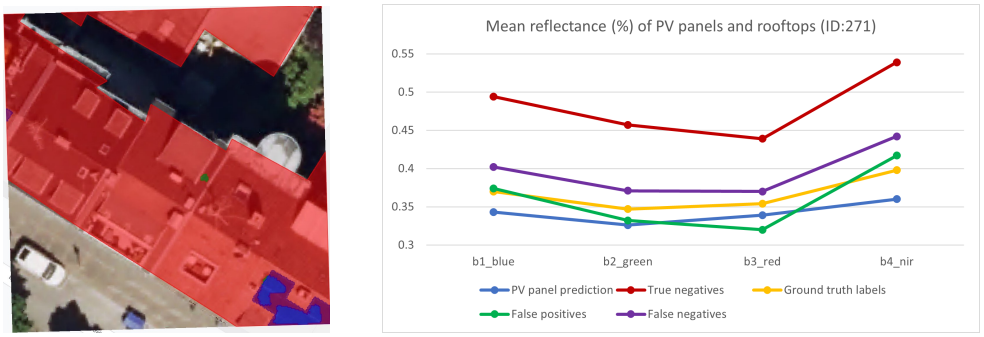


Figure 5.18.: RGB-NIR: Classification example of the city center; Special feature: **Black PV panels**

The next two examples present nearly successful classifications in the city center (Figure 5.19) and the suburbs (Figure 5.20). Both examples demonstrate how well classifications can work when homogeneous rooftops in terms of color, size, and structure, and PV panels in either dark or bright colors compose a clear contrast. This contrast is indicated by the distance between the MR of TN and ground truth.

Considering the MR of the NIR channel, it becomes evident that in both cases no specific pattern can be derived from the MR.

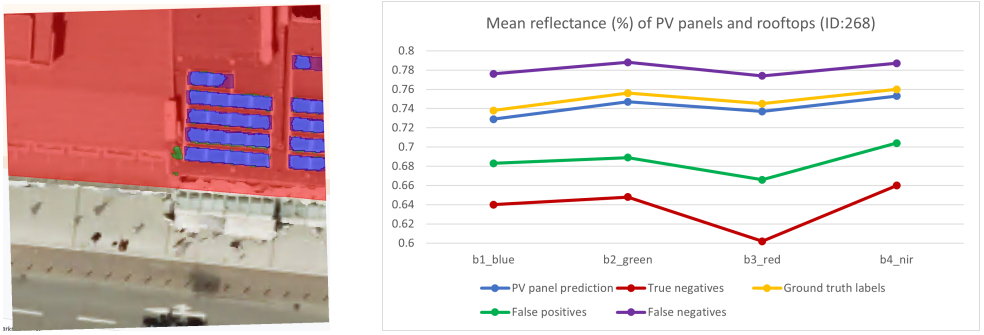


Figure 5.19.: RGB-NIR: City center example with blue PV panels

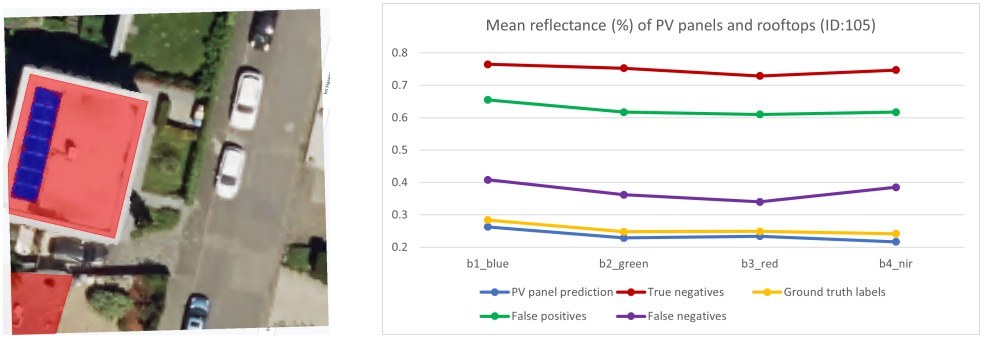


Figure 5.20.: RGB-NIR example with black PV panels in the suburbs

The visual differences of the last comparison are previously analyzed in Figure 5.7 and Figure 5.15. However, the MR of the FP provides a better understanding of the misclassification. While most PV panel predictions in previous examples show a flat curve of RGB-NIR reflectances, this curve in Figure 5.22 has a kink towards a lower NIR reflectance. This kink might be influenced by the FP prediction of the black rooftop in the lower left corner as well as of the STC. Likewise, the gap between ground truth reflectance and PV panel prediction widens.

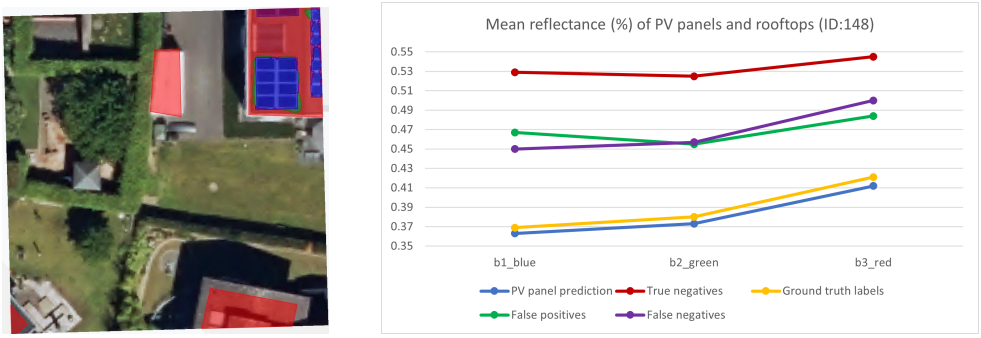


Figure 5.21.: RGB classification located in the suburbs; Special feature: **STC**

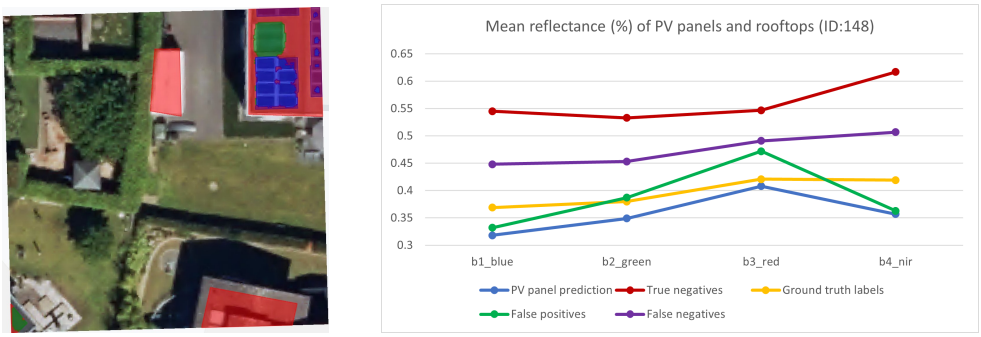


Figure 5.22.: RGB-NIR classification located in the suburbs; Special feature: **STC**

To gain a better understanding of the difference between PV panels and STC, the MR of only one building is extracted in the following Figure 5.23. The FP curve of the STC proves a strong mismatch to the ground truth label concerning the reflectance of the red channel. This reflectance is converging with the FN and TN curves, proving that the classification as a PV panel would be unlikely if the MR of the NIR channel did not converge to the ground truth label. Consequently, the NIR channel is mainly contributing to the confusion between PV panels and STC.

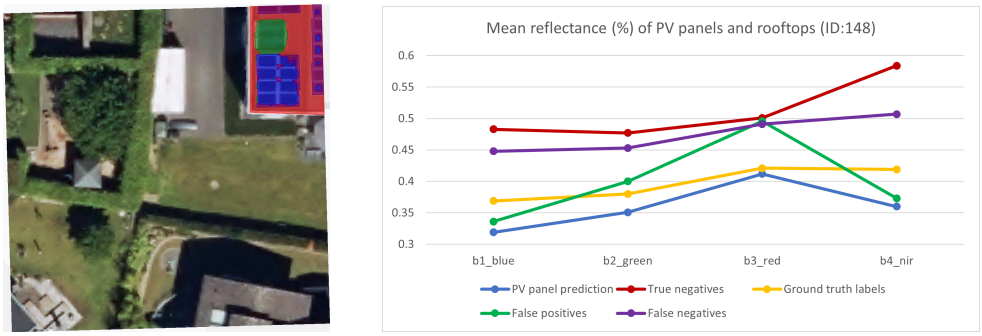


Figure 5.23.: RGB-NIR classification of one building in the suburbs; Special feature: **STC**

5.4. Classification of lower-resolution TrueDOPs

This subchapter is covering the last experiment carried out dealing with the classifications of TrueDOPs at 20 cm spatial resolution instead of 10 cm TrueDOPs utilized in previous classifications.

5.4.1. Quantitative evaluation of RGB classifications at 20 cm resolution

The classification scores in Table 5.4 show that the commercial area network performs better with lower-resolution images than networks trained on city center or suburb images. Most notable is the gap between precision and recall of city center and suburb classifications. It indicates a low number of FP predictions but an even higher number of FN predictions represented by missing PV panel predictions.

Area	precision (%)	recall (%)	F1-score (%)	IoU (%)
commercial	87.29	**85.17**	**86.22**	**86.89**
city center	**93.12**	12.47	22	55.4
suburbs	85.46	28.12	42.32	62.89
all areas	77.09	62.09	68.78	75.04

Table 5.4.: Evaluation of U-Net based on RGB TrueDOPs at 20 cm resolution

Despite the constant classification scores of the commercial area, Figure 5.24 demonstrates performance drops for lower-resolution images, particularly in the case of the city center and suburb networks.

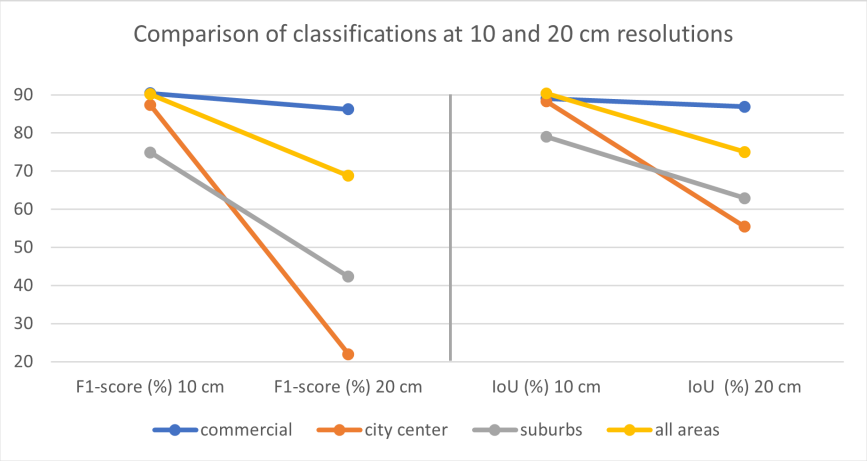
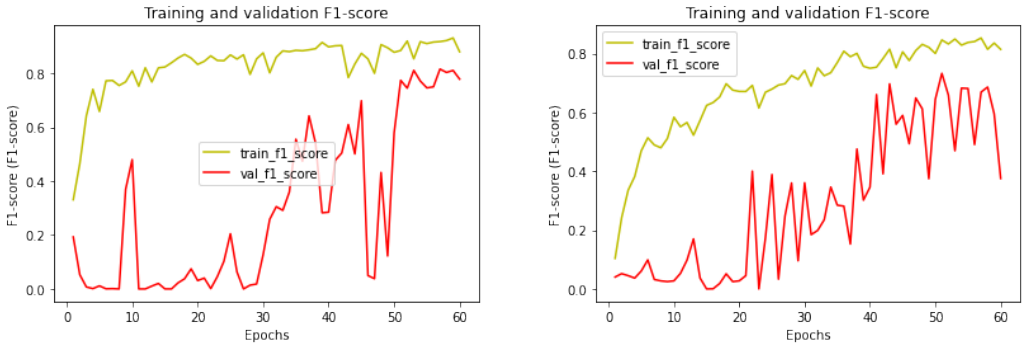


Figure 5.24.: Comparison of classifications at 10 and 20 cm resolutions assessed with F1-score and IoU

Nevertheless, both networks face difficulties when training on lower-resolution images (see Figure 5.25). Strong fluctuations in the performance curve (according to the F1-score) can be observed. After 50 epochs of training on images of the commercial area, the learning process starts to stabilize. In contrast, Figure 5.25(b) does not show a stabilization within the first 60 epochs of training the suburb network.



((a)) Commercial area ((b)) Suburbs

Figure 5.25.: Training and validation F1-score for each epoch

5.4.2. Visual evaluation of RGB classifications at 20 cm resolution

Commercial area. Lastly, one example from each subarea is presented. Although Figure 5.26 shows a prediction of a network trained on images from all areas, it demonstrates well how the contrast between the PV system and the rooftop color, and its size affect its predictions.

City center. In Figure 5.27, an image patch classified by the city center network provides an almost empty prediction mask. The predicted probability patch reveals many bright lines of low probabilities indicating the consideration of multiple objects by the network.

This output gives an impression of the difficulties faced in a heterogeneous urban area with a higher variety of objects, in particular for a larger image extent. Further, it partially contributes to the low classification score of city center image patches at 20 cm resolution.

Suburbs. Different from Figure 5.6, the skylight in the middle of the PV system is not considered as a FP. However, the glass roof of a carport is misclassified. Also, the predicted probability is showing the PV panel with less confidence than previous predictions but it is high enough to exceed the threshold of 0.5.

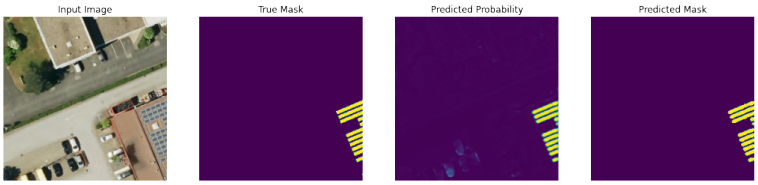


Figure 5.26.: Classification of a 20 cm resolution images of the commercial area

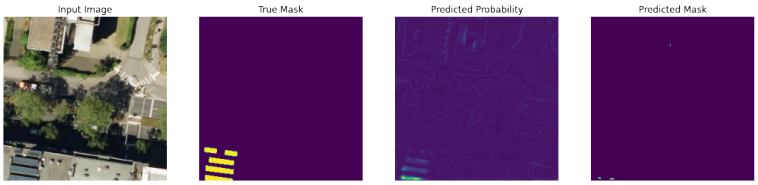


Figure 5.27.: Classification of a 20 cm resolution images of the city center

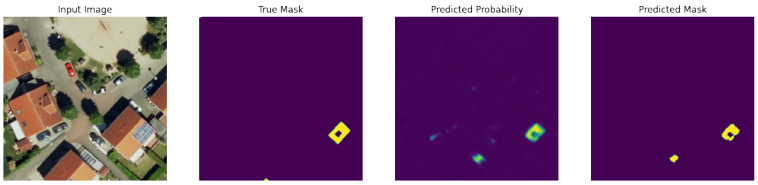


Figure 5.28.: Classification of a 20 cm resolution images of the suburbs; Special feature: **Carport**

6. Discussion, conclusion and future work

Following this research on how suitable a CNN with U-Net architecture is for the detection of PV panels in aerial images, the aim of emphasizing the significance of area-specific ground truth data is achieved as quantitative and qualitative analyses demonstrate in Chapter 5. In the context of the research questions and related research, these results are discussed in the following Section 6.1. The limitations encountered in this research are summarized in Section 6.2, followed by the conclusions on the research questions (Section 6.3). Further, the outcome contributing to current scientific knowledge is presented in Section 6.4. The final chapter provides an outlook of methods and analysis to be potentially implemented in future research.

6.1. Discussion

In this section, the preliminary results, defining the hyperparameters of the U-Net in Section 4.4.1, and the results achieved in Chapter 5 are discussed in the context of the results of related research described in Section 2.7 and the research questions defined in Section 1.3.

First, the hyperparameters are discussed, starting with the initialization of weights. The initialization of random weights using He uniform stands in contrast to the often used transfer-learning method in related research. The method is defined by the use of weights of another model that was trained on large data sets containing millions of images, such as ImageNet or COCO (ImageNet, 2020; Lin et al., 2014). In the case of this research, this method could be especially useful to avoid initial fluctuations of validation losses in the training, which are reinforced by little training data and small batch sizes. Transfer-learning could allow a head start in the training, which is reflected by a constant training progress inducing the model to converge faster, meaning that fewer epochs are required to train the model. For that purpose, Da Costa et al. (2021) implemented pre-trained weights from ImageNet, although its effect was not further examined. Nevertheless, the effect of implementing pre-trained weights is debatable. Whereas De Jong et al. (2020) concluded that the model benefits from transfer-learning, Malof et al. (2017) demonstrated that pre-trained weights are not a guarantee for an improvement of the model's performance.

Concerning the number of epochs, it is noticeable that most related projects are relying on a constant number of epochs rather than implementing the early-stopping method, despite the use of relatively high numbers of training images (De Jong et al., 2020; Malof et al., 2017). Finding an appropriate number of epochs without causing the model to overfit was especially challenging with few input images. The number of epochs is partially linked to the number of input images since more images can train a model much faster than fewer images. In comparison to this research, Castello et al. (2019) trained a U-Net on 4,680 images for 75 epochs and Da Costa et al. (2021) trained a U-Net on 210 images for a fixed number of 300 epochs, which is roughly the same number of images used in this research to train the model on all areas. Nevertheless, it became clear that the number of epochs significantly

depends on how well the model can detect PV panels, which is determined by the characteristics of the ground truth data but also by the size of the PV systems, in proportion to the image resolution and the image dimensions. Latter defines the proportion of target class pixels per image, which varies concerning different land use types.

Additionally, the proportion of target class pixels per image patch is of great importance for the loss function and the learning rate. Both hyperparameters are required to be selected with care regarding the data used. It turned out that the combination of the BCE and a learning rate of 0.0001 works best for the area-specific data sets, in comparison to the FL. In contrast, both Castello et al. (2019) and Da Costa et al. (2021) aimed to address class imbalances with weighted loss functions, such as the Dice Loss or the weighted pixel-wise categorical cross entropy function. This would distort the comparison between area-specific ground truth data since there is no class imbalance for commercial areas while there is an imbalance for suburbs. This research addresses this issue by carefully choosing image dimensions that consider PV system sizes of different land use types. Also, it was attempted to counter-act the effect of class imbalances by solely considering those image patches that contain target class pixels.

Nevertheless, the implementation of weighted loss functions can be a reasonable approach for image resolutions of less than 10 cm. As indicated by De Jong et al. (2020), the minor resolution difference between 10 and 20 cm impacts the model's ability to differentiate small objects, such as skylights, from PV panels. Similar results were obtained in Section 5.4.1 when the spatial resolution was decreased to 20 cm while keeping the same hyperparameters. In addition to FP classifications that confuse skylights or glass roofs with PV panels, the impact of heterogeneous environments (e.g., various types of urban objects, building shapes, and heights) became evident. This impact is reflected by significant performance drops of models that classified images of the city center.

It is hardly possible to compare the quantitative results of models that were trained on millions of images (De Jong et al., 2020; Malof et al., 2017) and this U-Net, which relied on 100-300 images when it comes to their significance. Nevertheless, similar challenges, as well as qualitative results, can be observed in related projects. In summary, the performance of the model trained on RGB images (10 cm) is quantified by an F1-score of 90.16% and an IoU of 90.36%. The results of subareas vary between an F1-score of 74.89% for the suburbs and 90.44% for the commercial area and IoU scores between 78.96% and 88.96% for the suburbs (lower score) and the commercial area (higher score), respectively. In comparison, this performance is significantly better than the U-Net's performance of Castello et al. (2019) achieving an F1-score of 80% and an IoU of 64%. Nevertheless, their model was applied to a greater variety of urban and rural settings from different regions. In contrast, Da Costa et al. (2021) achieved a better performance (F1-score: 95.38%; IoU: 91.17%), given that their project focuses solely on one type of PV system, namely large-scale solar plants.

Moreover, two observations concerning the precision and recall scores occurred similarly in related research. Firstly, the recall scores turned out to be lower than the precision scores in all experiments, except in the case of the commercial area, which achieved the highest recall of 91.5% (see Section 5.1.1). Da Costa et al. (2021) achieved a similar gap by obtaining a recall score of 93.1% and a precision score of 88.5%. It can be assumed that this gap was determined by the size of the PV systems (large-scale solar plants and large PV systems in commercial areas) in the images, resulting in fewer target class pixels being omitted. In the case of this research, the annotations of PV panels might contribute to a higher FP rate, resulting in a lower precision score. The PV panels were annotated in form of PV panel arrays, also in cases where there is barely a gap between the arrays. Having arrays annotated

individually aimed to collect specific meta data about their color, which can differ when the PV panels are directly exposed to the sun or slightly opposed. Especially smaller gap sizes tend to be generalized by the model as one continuous surface by filling the gaps with FP predictions, which affects the precision score (see Section B.2).

The second observation concerns a higher precision than recall score. The precision scores of classifications at 10 cm resolution varies between 89.1% and 97.86%. The gap between both scores is reflected by more FN than FP predictions, meaning that most PV panel classifications are correct, while few PV panels are not detected at all. A comparable discrepancy is observed in the results of Malof et al. (2017) achieving a recall of 80% and precision of 95%. The results of this research demonstrated that heterogeneous rooftops and PV systems in terms of rooftop sizes, shapes, and colors, as well as PV panel types, cause more FN predictions which affect the recall score.

Another effect that could contribute to a lower recall score is the misclassification of PV panels at the edges of image patches. To avoid such artifacts, Da Costa et al. (2021) followed the image mosaicking approach by Carvalho et al. (2021), in which overlapping image patches are generated to eliminate errors at the patches' edges. Although this method was not implemented in this research, a heat map showing all FN predictions (Figure 5.8) was created to analyze potential errors. The heat map proved that no artifact is systematically generated at the patches' edges.

Furthermore, this research reflects the large-scale cross-validations conducted in the DeepSolaris project by De Jong et al. (2020) on a local level (see Section 5.2). In the DeepSolaris project, the model's performance remained constant when the training area was nearby the validation area, while performance drops of the model were noticed for cross-validations within an entire state as well as in a cross-border context. Similarly, this thesis proved that differences in architectural and urban characteristics can already have an impact on the model's predictions within a city. In particular, this is the case between the suburbs and commercial areas.

It is important to note that the impact of the NIR channel on the detection of PV panels cannot be set in the context of related research as Da Costa et al. (2021) did not evaluate the impact of the NIR band. Moreover, there is no further research known that examines the use of the NIR channel in aerial imagery to detect PV panels. However, the results indicate mixed effects. Minor improvements and declines in the detection rate were noticed in the images of commercial areas and the city center, as well as in all images combined. In the case of the suburbs, the NIR rather caused a performance drop of the model than an improvement of the performance (see Section 5.3.1).

Overall, the implementation of the Adam optimizer, with a learning rate of 0.0001, and the BCE as a loss function was beneficial for conducting the experiments described in Section 4.5. The discussion of the hyperparameters and the corresponding results prove the importance to consider the correlation between hyperparameters and area-specific ground truth data. Therefore, improvements in the detection rate (in terms of the F1-score and IoU) can be achieved for different land use types, when adapting the model's hyperparameters such as loss function, learning rate, the number of epochs, or batch sizes with regard to the characteristics of the AOI and the PV systems. Finally, setting the quantitative and qualitative results in the context of related research indicates how well U-Net performed in multiple experiments.

6.2. Limitations

The experiments indicate great potential for detecting PV panels on rooftops while having certain limitations discussed in the following section.

Collecting ground truth data. Since the ground truth data is manually collected, its quality strongly depends on the annotator's ability to identify PV panels. As explained in Section 3.2, objects are only annotated if they are PV panels with high confidence to avoid feeding the model with false data. By doing so, PV panels that appear poor or ambiguous in the image are neglected as training data to not affect the detectability of PV panels. The annotator could overcome this limitation if accurate PV panel locations are available for the training area. Having a complete ground truth data set has the potential of improving the recall rate by omitting fewer PV panels while it might decrease the precision score due to more FP predictions since the model becomes less specific about the object of interest.

Amount of input data. Having little training and validation data, and therefore small batch sizes, reinforce fluctuations in the learning process. As a consequence, it takes longer to reach the point of convergence, meaning more epochs are needed to train the model appropriately. Also, a larger set of ground truth data allows a more comprehensive representation of reality, which increases the classification scores' significance. Against this backdrop, the results of the thesis must be considered in the context of limited data given.

Data augmentation. To analyze the correlation between rooftop and PV panel colors, the input images were not augmented in terms of brightness, contrast, saturation, or hue. Therefore, the basic data augmentation covering horizontal and vertical flips can be considered a limitation to the overall performance of the model.

Weights. The impact of utilizing randomly initialized weights or transfer-learning is not examined in this research. Therefore, it can be considered as a limitation as it is not known whether the use of random weights is affecting the learning process compared to the use of pre-trained weights. The choice to not implement transfer-learning in this thesis is based on the number of image channels used as input. Pre-trained weights are commonly trained on data sets consisting of 3 channel RGB images while this thesis incorporates 4 channel images consisting of RGB and NIR. Approaches to incorporate this method only for the RGB channels while initializing the weights for the NIR channel randomly would distort the comparability between RGB and RGB+NIR classifications.

Output format. A minor drawback of the workflow implemented for this thesis is the restriction to the PNG format instead of using TIFF. The restriction is determined by the available input formats offered by TensorFlow. This complicates the use of output prediction in geographical information systems (GISs). Nevertheless, a workaround to tackle this issue is presented in Section 4.6.

6.3. Conclusion

- What is the impact of different land use types on the detection of PV panels?

  The results of Section 5.1 demonstrate various aspects emerging from different land use types that need to be considered when compiling an appropriate training data set.

These aspects are variations in PV panels sizes and urban as well as architectural characteristics that impact the efficiency of the networks' learning processes and, therefore, the quality of the classification. Commercial areas stand out due to their homogeneity in terms of little variations in rooftop characteristics, such as roof colors, slopes, and sizes. This homogeneity in combination with predominantly large PV systems facilitates the training process of the network since it can converge faster for clear structures in which PV panels stand out noticeably due to greater coverage of target class pixels per image patch. However, the opposite effect emerges from residential areas in the suburbs having small PV systems installed on flat or pitched roofs with up to 6 different roof colors resulting in a low recall score caused by falsely classified PV panels indicating a class imbalance (see Section 4.2.2 and Section 5.1.1). Despite a high precision score, very specific objects can be picked out that cause potential confusion with PV panels, such as skylights, glass roofs, STC, and conservatories. Since a city center represents a mix of commercial and residential characteristics, it is most suitable as a training area for a network predicting PV panels in commercial areas and suburbs.

Overall, the experiments prove that it is of great relevance to adapt the training data to the properties of the AOI.

- Why is the correlation between roof color and panel color affecting the detection of PV panels?

  Regardless of the land use type, the results indicate that networks are prone to failure when black PV panels are installed on dark rooftops. In particular, this applies to black PV panels without a bright frame, which affects their detectability since they compose a continuous surface rather than clear patterns that are easier to recognize for CNNs. Having black rooftops and PV panels predominantly located in suburbs (in the case of this thesis) indicates an aspect contributing to a low recall score of 60.66% that impairs the detection rate (Section 5.1.1). On the contrary, a high contrast composition of PV panels and rooftops facilitates successful detections. Consequently, it is recommended to pay attention to the rooftop color as it can be considered an essential factor affecting the detection.

- What is the effect of adding near-infrared data to aerial images on the detection of PV panels?

  Adding a NIR channel to RGB imagery has indicated different effects on the detection of PV panels in different areas (see Section 5.3.1). While there is little to no effect on the detection process in the city center, there is only a marginal performance drop in the case of the commercial area (from an F1-score of 90.44% to 88.72%) as well as a slight increase in detection performance of all areas combined (from an F1-score of 90.16% to 91.75%). These performance changes are marginal, which is why they cannot be assigned to a specific cause. It can be assumed that these minor trends change when the model is repeatedly trained due to little training data. Most noticeable are the negative impacts on the detection rate of PV panels in the suburbs, where the F1-score declines from 74.89% to 68.21%. In this case, analyses have shown how the NIR channel contributes to misclassifications between PV panels and other objects. Nevertheless, the classification based on all areas (300 images in total) achieves the highest F1-score of all experiments of 91.75%.

- How sensitive is the model towards lower-resolution images with regard to the PV system size?

The DL model is particularly sensitive towards lower-resolution images of areas in which comparatively small PV systems are located (see Section 5.4). Accordingly, the sensitivity depends strongly on the ratio between the spatial resolution and the image patch dimensions. In this study, the spatial resolution is decreased from 10 to 20 cm per pixel while keeping the patch pixel dimensions to feed the U-Net with the same number of input pixels as for image patches of 10 cm resolution. Consequently, a scenery of a greater extent is shown to the network in which PV panels cover smaller portions of image patches represented by fewer pixels, which causes a greater imbalance between target class pixels and background information. Further, the characteristics of PV panels are less noticeable in the image making it more challenging for the model to learn them. Additionally, more objects are exposed to the model which might increase the risk of misclassification.

Conclusion on the main research question:

- To what extent is a CNN with U-Net architecture suitable for detecting PV panels on rooftops?

  This thesis aims to detect PV panels on rooftops using a CNN with U-Net architecture. Within the scope of the research questions, the outcomes provide insights into the suitability of a U-Net to detect PV panels. Multiple aspects need to be considered to allow successful classifications, such as defining ground truth data with urban and architectural characteristics corresponding to the area of predictions as well as appropriate hyperparameters concerning the ground truth data. The impacts of these characteristics are outlined by the subquestions indicating that a U-Net is overall suitable for classifying PV panels on RGB TrueDOPs at 10 cm spatial resolution.

6.4. Contribution

The contributions to scientific knowledge concerning semantic segmentations of PV panels through CNNs are summarized by the following categories.

Land use types. This thesis contributes to research by emphasizing the impact of differences in land use types and their characteristics on the detection of PV panels. It proved that urban and architectural differences within one city have a significant impact on the detection rate of a CNN with U-Net architecture. Therefore, it brings new knowledge to data-driven approaches concerning the detection of rooftop PV panels with CNNs.

Rooftop colors. The research conducted specific analyses to gain a better understanding of the correlation between rooftop color and PV panel types. This knowledge helps to interpret the recall and precision scores of semantic segmentations with CNNs.

NIR. The integration of NIR data in the training of a CNN for PV panel detection is an approach that has rarely been examined in related research. The results and analysis provided in this thesis indicate that adding a NIR to RGB imagery is not causing a significant improvement in the model's performance. Nevertheless, it might have potential when analyzed in semantic segmentations that are based on more training data.

Change of resolution. Comparing different sizes of rooftop PV systems in different urban environments showed how crucial the proportion between image dimensions, spatial resolution, and the PV system sizes is. The thesis presented an approach that considers the

rooftop sizes of the AOI in proportion to the image patch dimensions to find appropriate input dimensions. This helps in the prevention of class imbalances between target class pixels and background information.

6.5. Future work

Drawing on this thesis's limitations and insights gained during the work, recommendations and ideas about future research are presented in the following.

Additional input data. Future approaches to improve the classification of PV panels could make use of additional data. For instance, the use of height data or building footprint could put attention on the rooftop to reduce confusion with other urban objects in the surrounding.

Another approach could be the use of thermal infrared imagery, which recently gained popularity for monitoring the condition of PV panels. The promising results of Wang et al. (2022) and Buerhop et al. (2022) show how effective the use of thermal images can be in detecting PV panels in complex surroundings. Given the availability of georeferenced thermal imagery, which is often not available as open data, its usage as an additional image channel would outline an interesting data foundation for classifying PV panels in urban areas.

Classes. Since this thesis has focused on the detection of PV panels, a binary classification is employed. However, the implementation of a multi-class classification incorporating ground truth data of PV panels and STC could allow more precise learning to distinguish both types of panels.

Amount of training data. The process of collecting ground truth data is time-consuming and strongly depends on the annotator's knowledge of the PV system locations. As proven by Kriese et al. (2022) and Liu et al. (2020), enriching training data set with synthetic data generated by AI has great potential to increase the performance of CNNs. This method could help in future applications to train a CNN on larger data sets with greater varieties of PV panels.

Weights. Given the data basis of RGB imagery, it is recommended to make use of transfer learning to integrate pre-trained weights in the CNN. Nevertheless, it should be analyzed whether transfer-learning or random weights achieve better performances to obtain optimal results.

Regularization. Having a greater amount of data allows a more stable and, therefore, reliable learning process. Therefore, it is recommended to implement the method of early stopping to finish the training process at an appropriate number of epochs. If the method of early stopping is not implemented, it is of great importance to find the point of convergence manually to prevent the model from overfitting. It is inadvisable to rely on epoch numbers that were used in related research since the detectability of PV panels can be affected by the land use type, region, and country. In addition to batch normalization, a dropout method can be implemented to prevent the model from overfitting (Srivastava et al., 2014).

A. Reproducibility self-assessment

A.1. Marks for each of the criteria

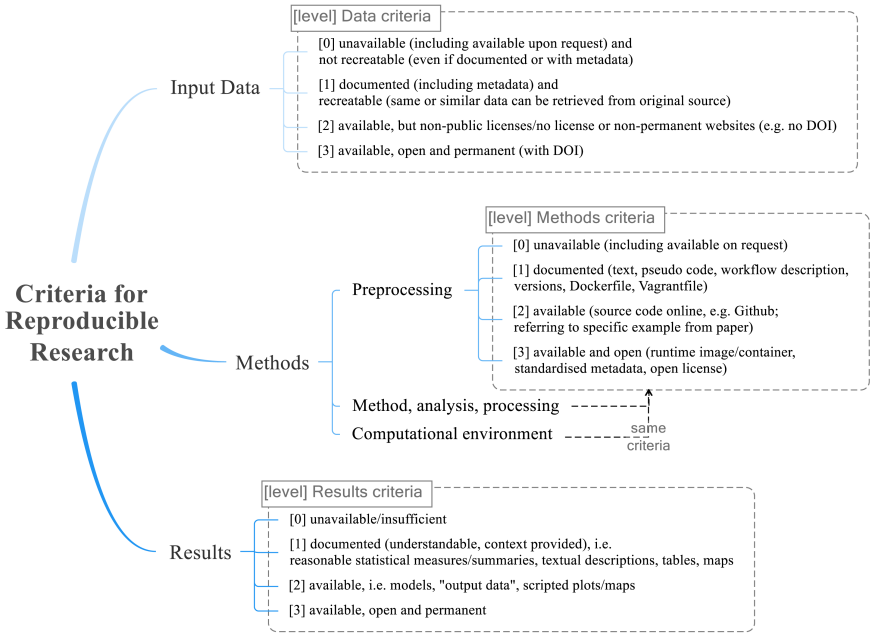


Figure A.1.: Reproducibility criteria to be assessed.

Grade/evaluate yourself for the 5 criteria (giving 0/1/2/3 for each):

no.	criteria	**grade**
1. Input data	Aerial images	**3**
	Ground truth data	**0**
	Building footprints	**3**
	Land use maps	**3**
2. Methods	Preprocessing	**1**
	Analysis	**1**
	Computational environment	**2**
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.

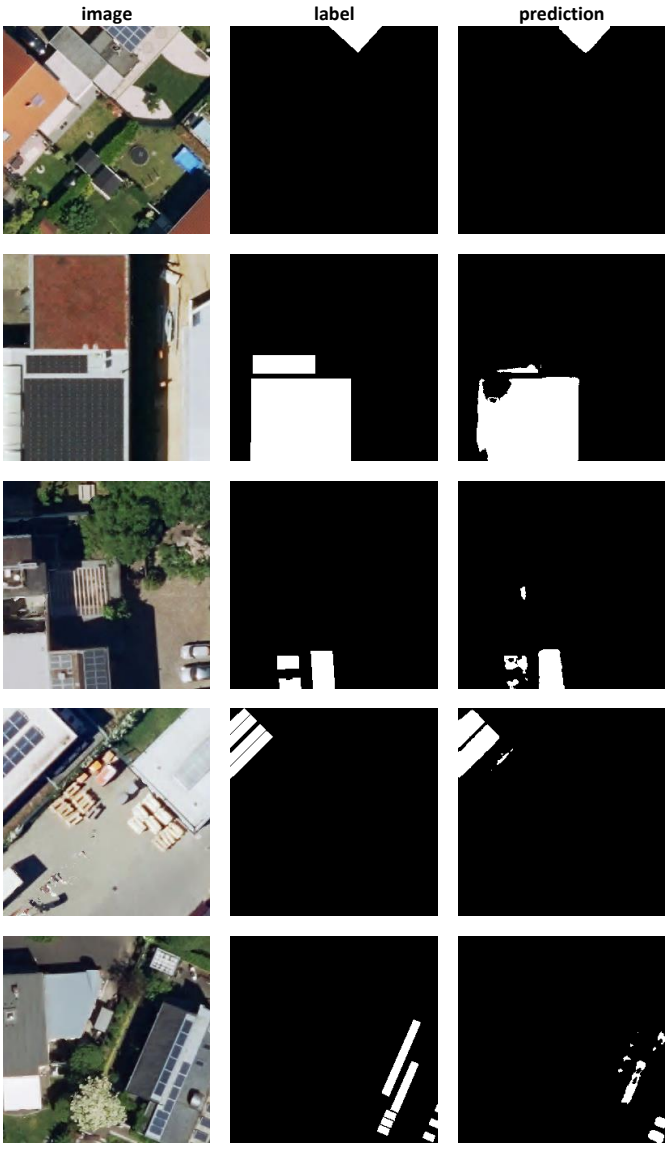
The first criterion concerns the training data, the building footprints, and the land use maps. The aerial images are permanently available in the SDI of NRW, see GeobasisNRW (2023). Similarly, the building and land use maps are openly available, see OpenNRW (2022b) and OpenNRW (2022b). In contrast, the ground truth data is manually generated and not publicly available.

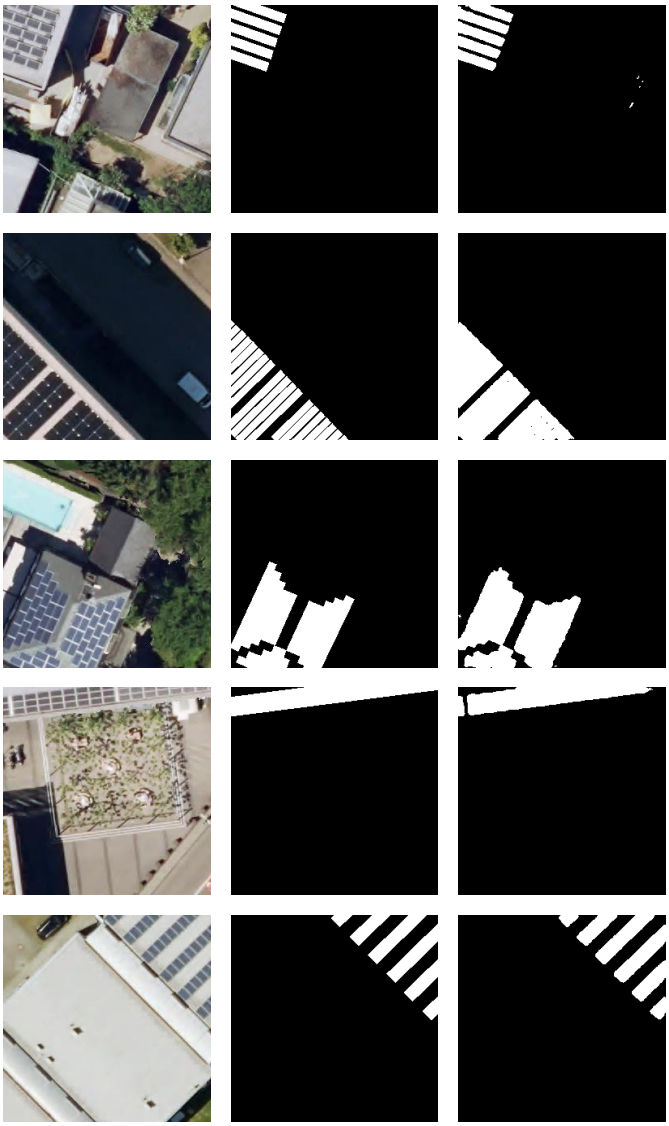
The implemented preprocessing steps and analyses are reproducible by following the documentation given by this thesis in Chapter 3 and Chapter 4. The computational environment concerns mainly free usable software, such as QGIS, and programming languages like R (in R-Studio) and Python (in Visual Studio Code). Additionally, the programming language JavaScript was used in the computing platform GEE, which is free for noncommercial purposes. The most important environment for carrying out the work was the Colab from Google Research in combination with the ML library TensorFlow. Access to Colab is free. Nevertheless, to have constant access to the computing resources of the GPUs, 100 compute units were acquired for a fee of 9.25 €.

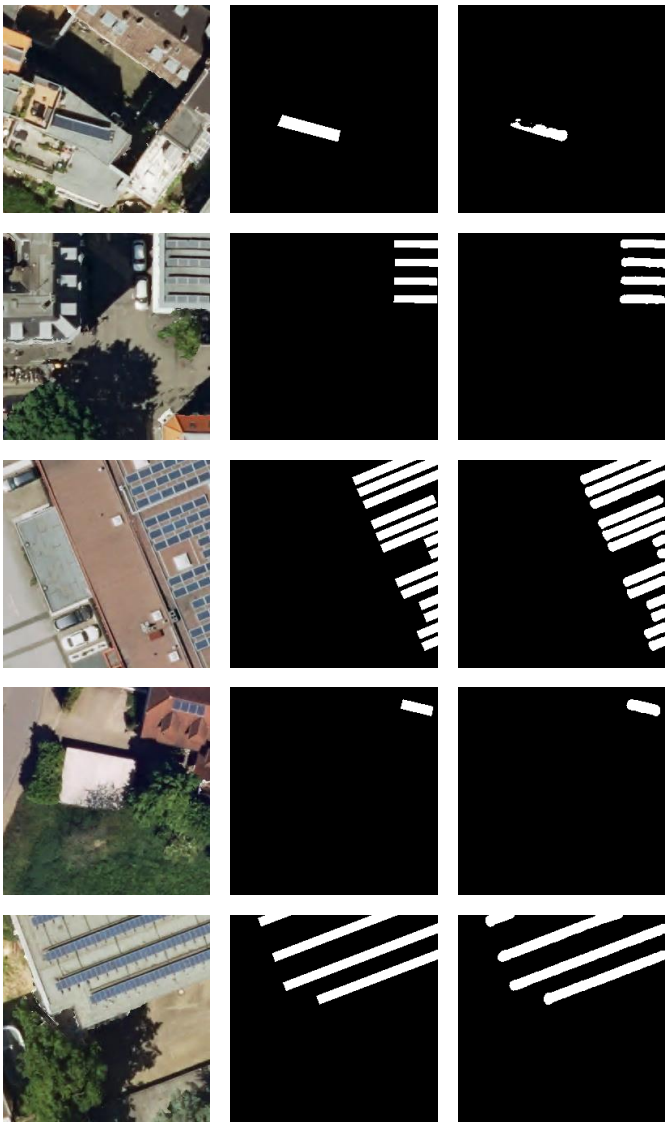
Lastly, the results and analyses are documented in form of descriptions, tables, and maps in Chapter 5. Additional output images that were used to compute a heat map in Section 5.1.1 are provided in the Appendix B. Moreover, samples from the output generated in cross-validation experiments are summarized in Appendix C.

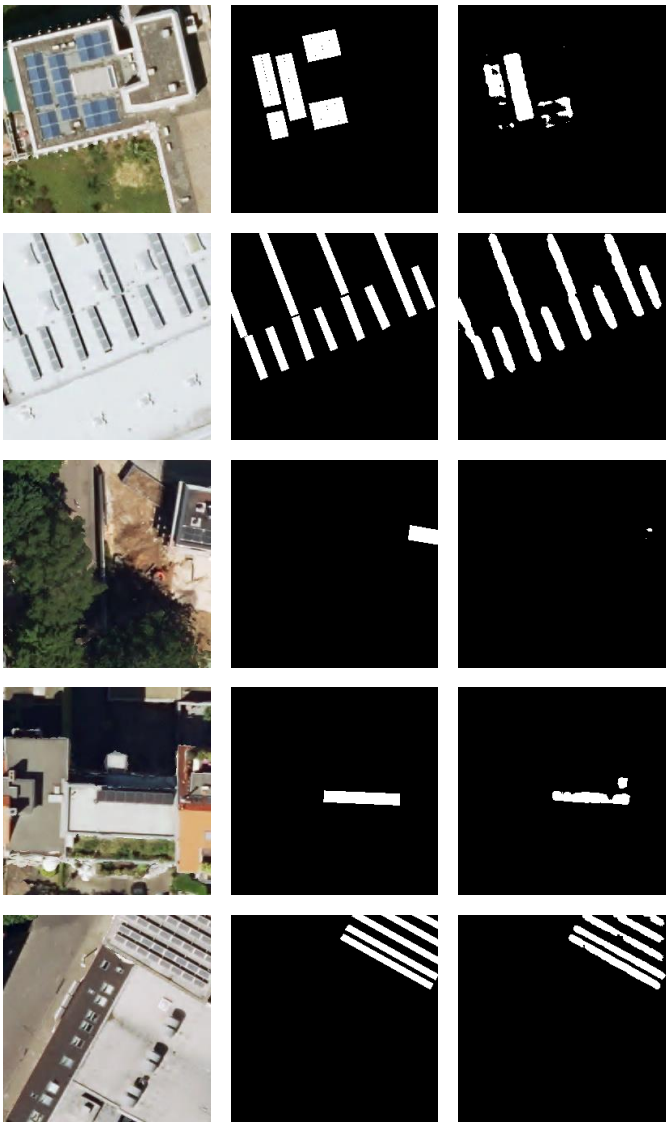
B. Images, labels, and predicted masks included for heat map

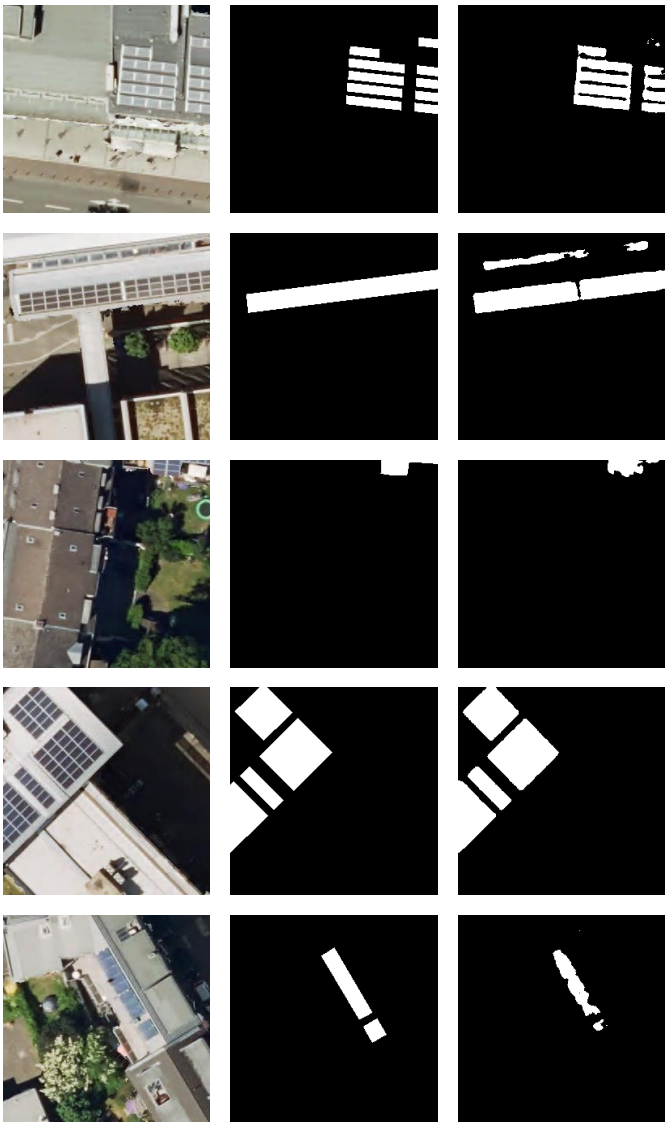
B.1. Patches from all areas











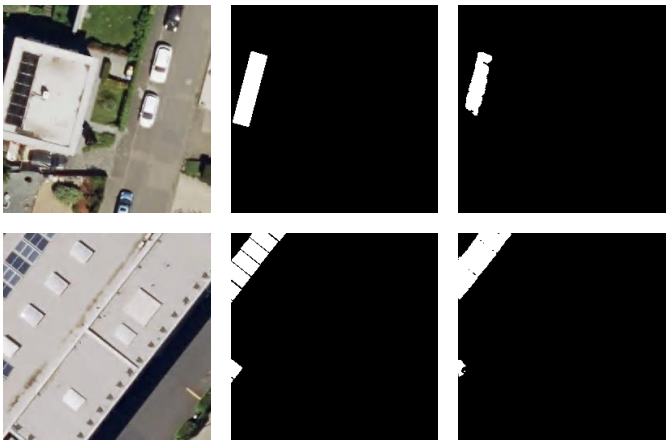
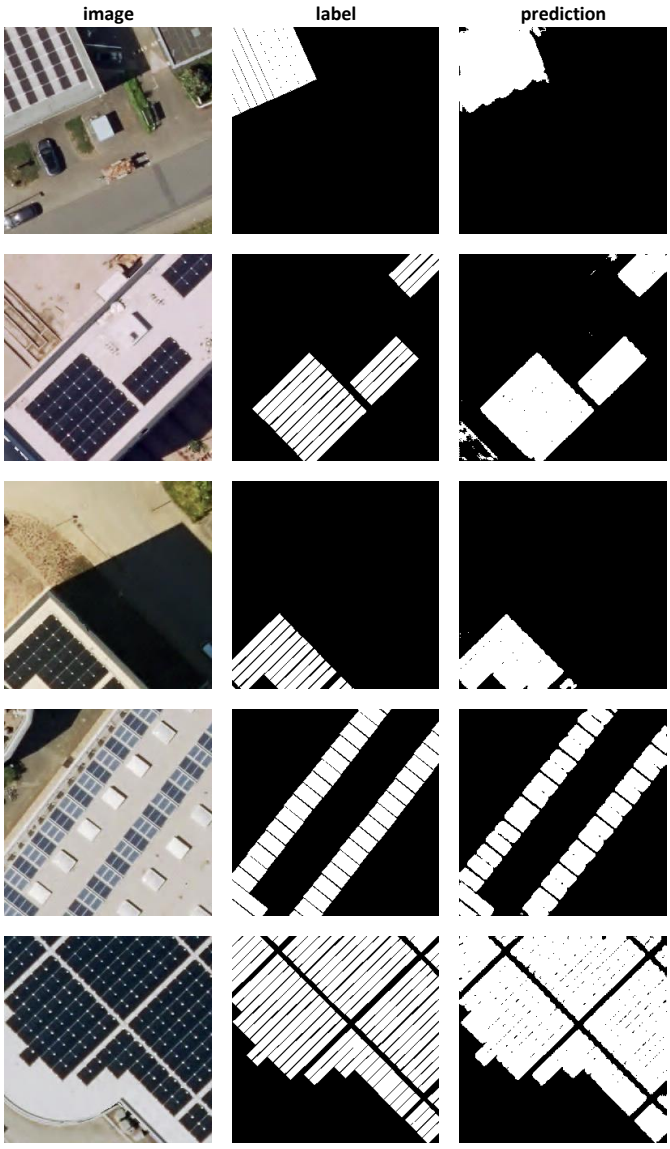


Figure B.1.: All subareas: RGB testing images, labels, and prediction

B. Images, labels, and predicted masks included for heat map

B.2. Model for commercial area



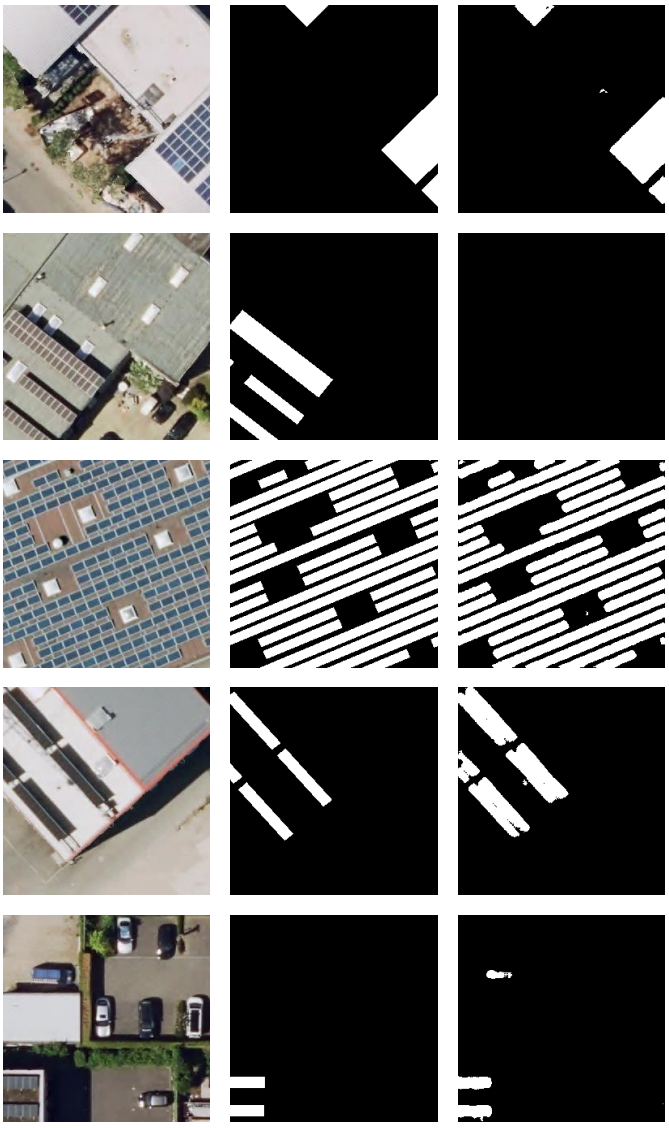
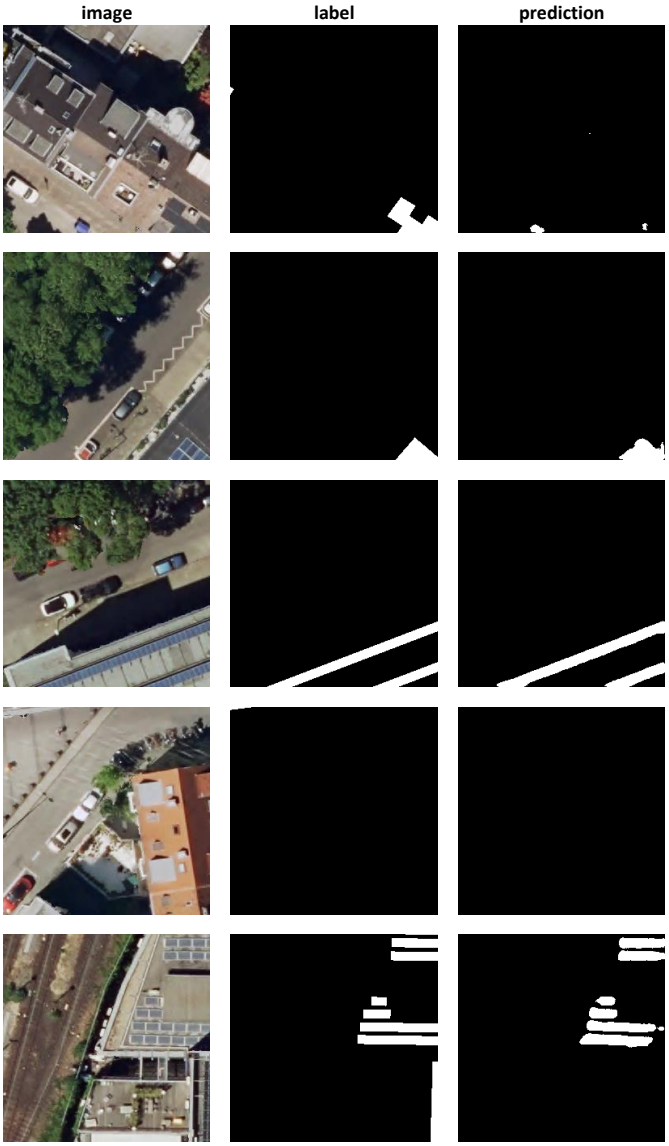


Figure B.2.: Commercial area: RGB testing images, labels, and prediction

B.3. Patches from the city center



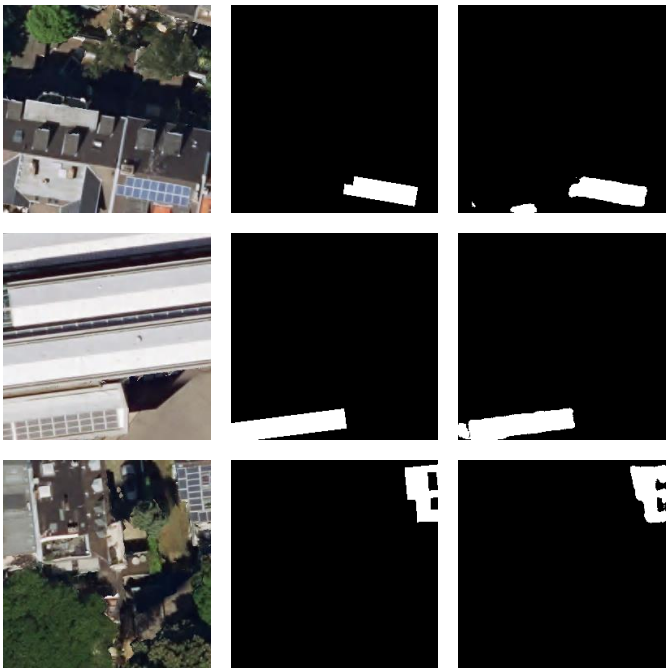
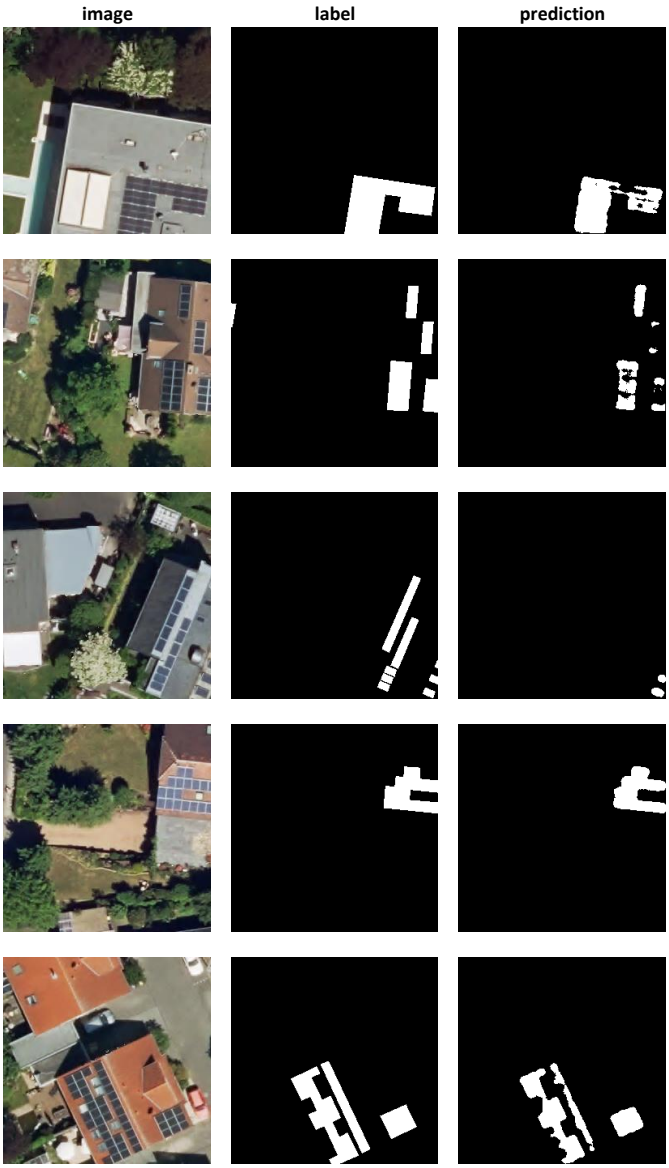


Figure B.3.: City center: RGB testing images, labels, and prediction

B. Images, labels, and predicted masks included for heat map

B.4. Patches from the suburbs



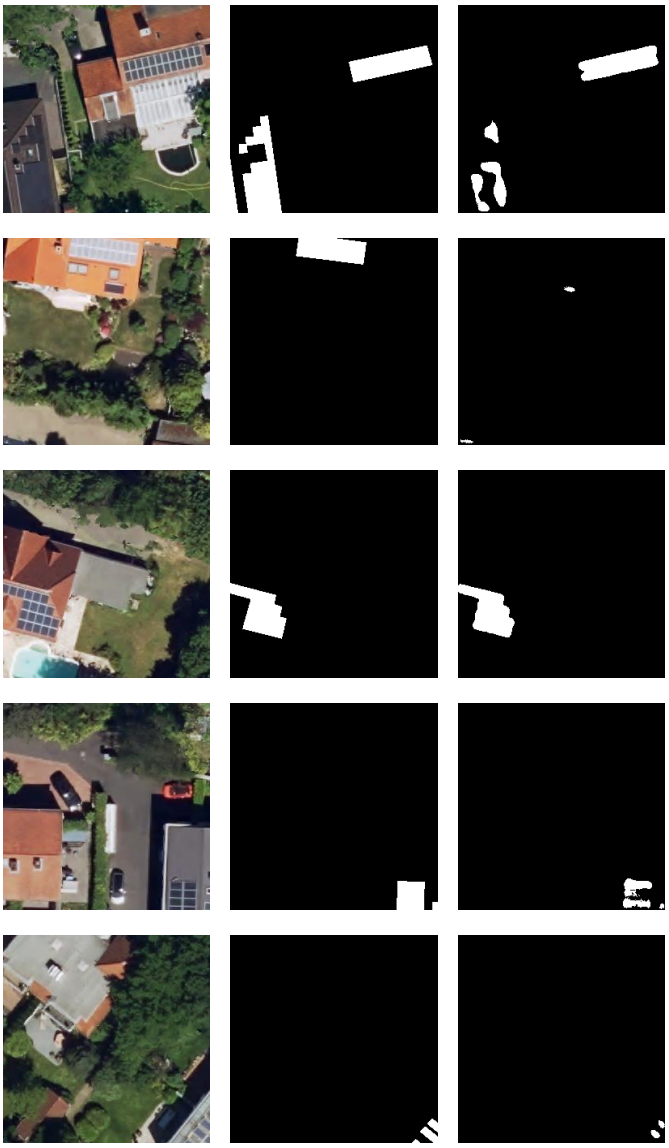


Figure B.4.: Suburbs: RGB testing images, labels, and prediction

C. Sample results of cross-validation

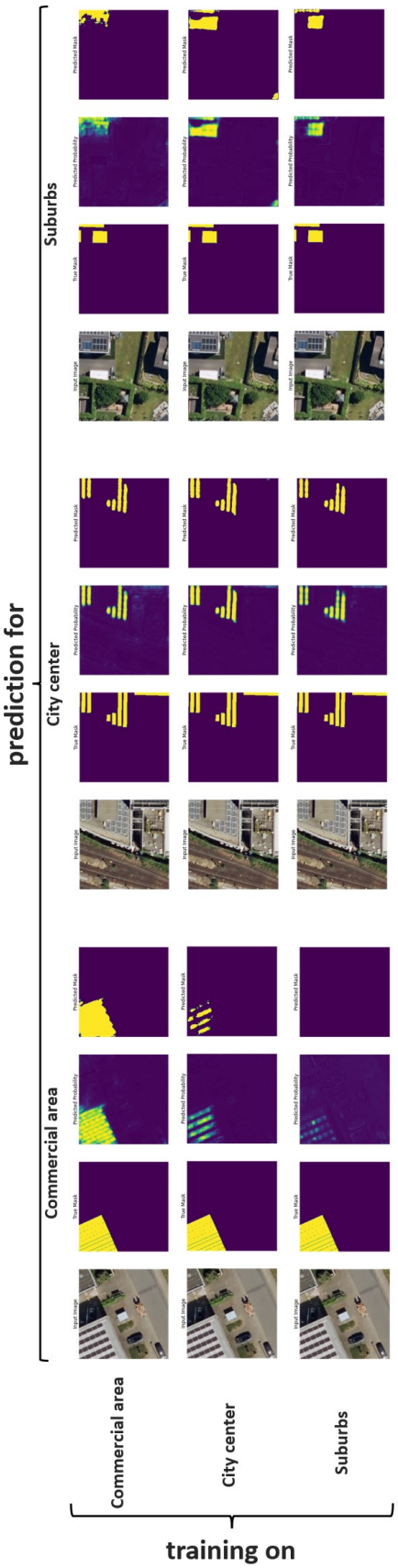


Figure C.1.: Cross-validation examples showing RGB testing images, labels, predicted probabilities, and prediction masks

Bibliography

Alam, M., Wang, J. F., Guangpei, C., Yunrong, L., and Chen, Y. (2021). Convolutional Neural Network for the Semantic Segmentation of Remote Sensing Images. *Mobile Networks and Applications*, 26(1):200–215.

Bishop, C. M. (1998). Neural networks and their applications. *Review of Scientific Instruments*, 65(6):1803.

Bottou, L. (1991). Stochastic gradient learning in neural networks. In *Proceedings of Neuro-Nîmes 91*, Nimes, France. EC2.

Bradbury, K., Saboo, R., Johnson, T. L., Malof, J. M., Devarajan, A., Zhang, W., Collins, L. M., and Newell, R. G. (2016). Distributed solar photovoltaic array location and extent dataset for remote sensing object identification. *Scientific Data*, 3(1):160106.

Buerhop, C., Bommes, L., Schlipf, J., Pickel, T., Fladung, A., and Peters, I. M. (2022). Infrared imaging of photovoltaic modules: a review of the state of the art and future challenges facing gigawatt photovoltaic power stations. *Progress in Energy*, 4(4):042010.

Carvalho, O. L. F. d., de Carvalho Júnior, O. A., Albuquerque, A. O. d., Bem, P. P. d., Silva, C. R., Ferreira, P. H. G., Moura, R. d. S. d., Gomes, R. A. T., Guimarães, R. F., and Borges, D. L. (2021). Instance segmentation for large, multi-channel remote sensing imagery using mask-rcnn and a mosaicking approach. *Remote Sensing*, 13(1).

Castello, R., Roquette, S., Esguerra, M., Guerra, A., and Scartezzini, J.-L. (2019). Deep learning in the built environment: automatic detection of rooftop solar panels using convolutional neural networks. *Journal of Physics: Conference Series*, 1343:012034.

Chen, D.-Y., Peng, L., Zhang, W.-Y., Wang, Y.-D., and Yang, L.-N. (2022). Research on self-supervised building information extraction with high-resolution remote sensing images for photovoltaic potential evaluation. *Remote Sensing*, 14(21).

Chen, X., Xiang, S., Liu, C. L., and Pan, C. H. (2014). Vehicle detection in satellite images by hybrid deep convolutional neural networks. *IEEE Geoscience and Remote Sensing Letters*, 11(10):1797–1801.

ClimateWatch (2022). Climate watch historical country greenhouse gas emissions data. https://www.climatewatchdata.org/ghg-emissions?end_year=2019&start_year=1990. Accessed: 2023-01-04.

Curier, R. L., Jong, T. J. A. D., Strauch, K., Cramer, K., Rosenski, N., Schartner, C., Debusschere, M., Ziemons, H., Iren, D., and Bromuri, S. (2018). Monitoring spatial sustainable development: Semi-automated analysis of satellite and aerial images for energy transition and sustainability indicators. *CoRR*, abs/1810.04881.

Da Costa, M. V. C. V., de Carvalho, O. L. F., Orlandi, A. G., Hirata, I., de Albuquerque, A. O., Silva, F. V. e., Guimarães, R. F., Gomes, R. A. T., and Júnior, O. A. d. C. (2021). Remote Sensing for Monitoring Photovoltaic Solar Plants in Brazil Using Deep Semantic Segmentation. *Energies 2021, Vol. 14, Page 2960*, 14(10):2960.

De Jong, T., Bromuri, S., Chang, X., Debusschere, M., Rosenski, N., Schartner, C., Strauch, K., Boehmer, M., and Curier, L. (2020). Monitoring spatial sustainable development: semi-automated analysis of satellite and aerial images for energy transition and sustainability indicators. *CoRR*, abs/2009.05738.

Deng, L. and Yu, D. (2014). Deep Learning: Methods and Applications. *Foundations and Trends® in Signal Processing*, 7(3–4):197–387.

EC (2022). Solar energy. https://energy.ec.europa.eu/topics/renewable-energy/solar-energy_en. Accessed: 2023-01-04.

Foody, G. M. (2017). Impacts of Sample Design for Validation Data on the Accuracy of Feedforward Neural Network Classification.

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4):193–202.

GeobasisNRW (2020). True orthophotos - keine verdeckungen durch umklappen mehr! https://www.bezreg-koeln.nrw.de/brk_internet/true_orthophotos/index.html. Accessed: 2022-12-07.

GeobasisNRW (2022). Digitale orthophotos. https://www.bezreg-koeln.nrw.de/brk_internet/geobasis/luftbildinformationen/aktuell/digitale_orthophotos/index.html. Accessed: 2022-12-07.

GeobasisNRW (2023). Digitale orthophotos (10-fache kompression) - paketierung: Einzelkacheln. https://www.opengeodata.nrw.de/produkte/geobasis/lusat/dop/dop_jp2_f10/. Accessed: 2023-01-11.

Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In Teh, Y. W. and Titterington, M., editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 249–256, Chia Laguna Resort, Sardinia, Italy. PMLR.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification.

IBM (2020). Gradient descent. https://www.ibm.com/cloud/learn/gradient-descent. Accessed: 2022-12-07.

IBM (2021). Overfitting. https://www.ibm.com/cloud/learn/overfitting. Accessed: 2022-12-07.

ImageNet (2020). About imagenet. https://www.image-net.org/about.php. Accessed: 2022-12-07.

Ioffe, S. and Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *32nd International Conference on Machine Learning, ICML 2015*, 1:448–456.

Kingma, D. P. and Ba, J. L. (2014). Adam: A Method for Stochastic Optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*.

Kriese, J., Hoeser, T., Asam, S., Kacic, P., Da Ponte, E., and Gessner, U. (2022). Deep learning on synthetic data enables the automatic identification of deficient forested windbreaks in the paraguayan chaco. *Remote Sensing*, 14(17).

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2017). ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM*, 60(6):84–90.

Lecun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature 2015 521:7553*, 521(7553):436–444.

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Lin, T. Y., Goyal, P., Girshick, R., He, K., and Dollar, P. (2017). Focal Loss for Dense Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):318–327.

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft COCO: Common Objects in Context. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8693 LNCS(PART 5):740–755.

Liu, W., Liu, J., and Luo, B. (2020). Can synthetic data improve object detection results for remote sensing images? *arXiv preprint arXiv:2006.05015*.

Malof, J., Collins, L., and Bradbury, K. (2017). A deep convolutional neural network, with pre-training, for solar photovoltaic array detection in aerial imagery. pages 874–877.

MaStR (2023). Zielsetzung des marktstammdatenregister.

Müller, D., Soto-Rey, I., and Kramer, F. (2022). Towards a guideline for evaluation metrics in medical image segmentation.

OpenNRW (2022a). Flaechennutzungsplan. https://open.nrw/dataset/flaechennutzungsplan-k. Accessed: 2022-12-07.

OpenNRW (2022b). Hausumringe nw. https://open.nrw/dataset/3f08a580-48ec-43c1-936d-d62f89c21cc9. Accessed: 2022-12-07.

O'Shea, K. and Nash, R. (2015). An Introduction to Convolutional Neural Networks.

PDOK (2022). Dataset: Luchtfoto / pdok (open). https://www.pdok.nl/introductie/-/article/luchtfoto-pdok. Accessed: 2022-12-07.

Pierdicca, R., Malinverni, E., Piccinini, F., Paolanti, M., Felicetti, A., and Zingaretti, P. (2018). Deep convolutional neural network for automatic detection of damaged photovoltaic cells. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2:893–900.

Rausch, B., Mayer, K., Arlt, M., Gust, G., Staudt, P., Weinhardt, C., Neumann, D., and Rajagopal, R. (2020). An enriched automated PV registry: Combining image recognition and 3d building data. *CoRR*, abs/2012.03690.

Reynolds, A. H. (2019). Convolutional neural networks (cnns). https://anhreynolds.com/blogs/cnn.html. Accessed: 2022-12-07.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597.

Ruder, S. (2016). An overview of gradient descent optimization algorithms.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature 1986 323:6088*, 323(6088):533–536.

Sharma, S., Sharma, S., and Athaiya, A. (2020). ACTIVATION FUNCTIONS IN NEURAL NETWORKS. *International Journal of Engineering Applied Sciences and Technology*, 04:310–316.

Shi, Q., Liu, X., and Li, X. (2017). Road Detection from Remote Sensing Images by Generative Adversarial Networks. *IEEE Access*, 6:25486–25494.

Simonyan, K. and Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*.

SolarReviews (2021). Types of solar panels: which one is the best choice? https://www.solarreviews.com/blog/pros-and-cons-of-monocrystalline-vs-polycrystalline-solar-panels. Accessed: 2023-01-05.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958.

UN (2022). What is renewable energy? https://www.un.org/en/climatechange/what-is-renewable-energy. Accessed: 2023-01-04.

UNFCCC (2015). PARIS AGREEMENT.

Vakalopoulou, M., Karantzalos, K., Komodakis, N., and Paragios, N. (2015). Building detection in very high resolution multispectral data with deep learning features. *International Geoscience and Remote Sensing Symposium (IGARSS)*, 2015-November:1873–1876.

Wang, J. L., Li, A. Y., Huang, M., Ibrahim, A. K., Zhuang, H., and Ali, A. M. (2019). Classification of White Blood Cells with PatternNet-fused Ensemble of Convolutional Neural Networks (PECNN). *2018 IEEE International Symposium on Signal Processing and Information Technology, ISSPIT 2018*, pages 325–330.

Wang, X., Yang, W., Qin, B., Wei, K., Ma, Y., and Zhang, D. (2022). Intelligent monitoring of photovoltaic panels based on infrared detection. *Energy Reports*, 8:5005–5015.

Xu, C. and Wang, H. (2022). Research on a Convolution Kernel Initialization Method for Speeding Up the Convergence of CNN. *Applied Sciences 2022, Vol. 12, Page 633*, 12(2):633.

Zeiler, M. D. and Fergus, R. (2013). Visualizing and Understanding Convolutional Networks. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8689 LNCS(PART 1):818–833.

Zhang, T. and Yu, B. (2005). Boosting with early stopping: Convergence and consistency. *Annals of Statistics*, 33(4):1538–1579.

Ševo, I. and Avramović, A. (2016). Convolutional neural network based automatic object detection on aerial images. *IEEE Geoscience and Remote Sensing Letters*, 13(5):740–744.

Colophon

This document was typeset using LaTeX, using the KOMA-Script class `scrbook`. The main font is Palatino.