



Diese Arbeit wurde vorgelegt am  
Lehrstuhl für Informatik 5 (Informationssysteme und Datenbanken)

## **Bildgestützte Identifikation von Gebäuden zur regionalen Bestimmung des Energiebedarfs**

Image-based identification of buildings for regional determination of energy demand

**Bachelorarbeit**

von

**Herrn Daniel Zelenak**

1. Prüfer: Univ.-Prof. Dr. Stefan Decker

2. Prüfer: Univ.-Prof. Dr. Andreas Ulbig

Betreuer: Klemens Schumann, M.Sc.

Luis Böttcher, M.Sc.

Philipp Hälsig, M.Sc.

Aachen, 04. November 2021

# Eidesstattliche Versicherung

Zelenak, Daniel \_\_\_\_\_ 394425 \_\_\_\_\_  
Name, Vorname Matrikelnummer

Ich versichere hiermit an Eides Statt, dass ich die vorliegende Bachelorarbeit mit dem Titel

## **Bildgestützte Identifikation von Gebäuden zur regionalen Bestimmung des Energiebedarfs**

selbstständig und ohne unzulässige fremde Hilfe (insbes. akademisches Ghostwriting) erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt. Für den Fall, dass die Arbeit zusätzlich auf einem Datenträger eingereicht wird, erkläre ich, dass die schriftliche und die elektronische Form vollständig übereinstimmen. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Aachen, 04. November 2021 \_\_\_\_\_  
Ort, Datum Unterschrift

### **Belehrung:**

#### **§ 156 StGB: Falsche Versicherung an Eides Statt**

Wer vor einer zur Abnahme einer Versicherung an Eides Statt zuständigen Behörde eine solche Versicherung falsch abgibt oder unter Berufung auf eine solche Versicherung falsch aussagt, wird mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft.

#### **§ 161 StGB: Fahrlässiger Falscheid; fahrlässige falsche Versicherung an Eides Statt**

(1) Wenn eine der in §§ 154 bis 156 bezeichneten Handlungen aus Fahrlässigkeit begangen worden ist, so tritt Freiheitsstrafe bis zu einem Jahr oder Geldstrafe ein.

(2) Straflosigkeit tritt ein, wenn der Täter die falsche Angabe rechtzeitig berichtet. Die Vorschriften des § 158 Abs. 2 und 3 gelten entsprechend.

Die vorstehende Belehrung habe ich zur Kenntnis genommen:

Aachen, 04. November 2021 \_\_\_\_\_  
Ort, Datum Unterschrift

# Kurzfassung

Durch den Klimawandel werden zukünftig vermehrt neue Technologien, wie erneuerbare Energien oder die Elektromobilität, eingesetzt. Die neuen Stromerzeuger und Lasten führen zu Herausforderungen (z.B. den Verteilnetzausbau) und bieten Potentiale, die in aktuellen Forschungsfragen untersucht werden. Diese Forschungsfragen benötigen hochauflösende räumliche Datensätze, die den Energiebedarf in zu analysierenden Regionen abbilden. Jedoch sind diese Daten oftmals nicht zugänglich (z. B. aufgrund des Datenschutzes). Eine zugängliche Datenquelle sind Luftbildaufnahmen. Mit diesen kann der Energiebedarf auch in Regionen untersucht werden, für die keine oder wenige Informationen zur Verfügung stehen.

Im Rahmen dieser Arbeit wurde deshalb eine Methodik entwickelt, die einen solchen Datensatz aufgrund von Luftbildaufnahmen generieren kann. Der Gesamtprozess wurde in zwei logische Abschnitte aufgeteilt: Die bildgestützte Gebäudesegmentierung im ersten und die Bestimmung des Energiebedarfes auf Grundlage der vorangegangenen Ergebnisse im zweiten Teil.

Im ersten Teil werden mit neuronalen Netzen Modelle zur Segmentierung von Gebäuden aus digitalen Orthofotos entwickelt. Zum Training der neuronalen Netze wird das Framework RasterVision verwendet. Mit OpenStreetMap und ALKIS als Trainingsgrundlage ist es möglich Modelle zu trainieren, die die Gebäudeumrisse segmentieren. Darüber hinaus können spezialisierte Modelle sogar zwischen Gebäudebauweisen unterschieden.

Im zweiten Teil des entwickelten Verfahrens werden auf Basis der segmentierten Gebäudepolygone Energiebedarfe abgeleitet. Dazu werden zunächst alle Gebäude mit einem RandomForest-Klassifikator einer Bauweise zugewiesen.

Gebäude sind stets Bestandteil eines Baublocks, dem wiederum ein energetischer Stadtraumtyp (EST) zugeordnet werden kann. Mit Hilfe eines K-Means-Clusterings werden den Blöcken passende EST-Klassen zugeordnet. Die EST-Klassen geben abschließend eine Auskunft darüber, wie viele Einwohner in einem Gebäude leben und welcher jährliche Energiebedarf pro Einwohner oder pro Quadratmeter Grundfläche zu erwarten ist.

In exemplarischen Untersuchungen wurden verschiedene Parameter und Verfahrensanpassungen untersucht. So konnte gezeigt werden, dass OpenStreetMap als Trainingsgrundlage zur Bilderkennung von Gebäude ausreichend ist. Dabei konnten auch Bilder unterschiedlicher Auflösung zwischen Training und Anwendung genutzt werden. Es wurde zudem festgestellt, dass es teilweise möglich ist, die Gebäude bereits anhand von neuronalen Netzen in verschiedene Bauweisen zu klassifizieren.

Das finale Verfahren wurde abschließend auf die Region Aachen exemplarisch angewendet und die Ergebnisse mit weniger granularen Referenzdatensätzen bzgl. Einwohnerzahl und elektrischen Energiebedarf verglichen. Dabei konnte gezeigt werden, dass die Gesamteinwohnerzahl mit einer Abweichung von 22% und der elektrische Energiebedarf mit einer Abweichung von 8% bestimmt werden konnte.

# Abstract

Due to climate change, the use of new technologies such as renewable energies or electromobility will increase in the future. The new power generators and loads lead to challenges (e.g. distribution grid expansion) and offer potentials that are being investigated in current research questions. These research questions require high-resolution spatial data sets that map the energy demand in a specific region. However, these data are often not accessible (because of privacy concerns). One accessible data source is aerial photography. With this, the energy demand can be analysed in regions for which no or little information on energy demand is available.

In the context of this work, a method was developed that can generate such a data set based on aerial photography. The overall process was divided into two steps: The image-based building segmentation is the first one, and the determination of the energy demand, based on the previous results, is the second part.

In the first part, neural networks are used to develop models for segmenting buildings from digital orthophotos. To train the neural networks the framework RasterVision is used. Using OpenStreetMap and ALKIS as a training basis, it is possible to train models that segment building outlines. Furthermore, specialised models can even distinguish building types.

In the second part of the developed procedure, energy demands are derived by utilizing the segmented building polygons. First, all buildings are assigned to a construction type using a RandomForest classifier.

Buildings are always part of a building block, which in turn can be assigned to an energetic urban space type (EST). With the help of a K-Means clustering, suitable EST classes are assigned to the blocks. Finally, the EST classes provide information on how many inhabitants live in a building and the annual energy demand per inhabitant or per square metre of floor space that can be expected.

In exemplary investigations, various parameters and process adjustments were examined. It was shown that OpenStreetMap is sufficient as a training basis for the image recognition of buildings. It was also possible to use images with different resolutions between training and application. Further, it was found that it is partly possible to classify buildings into different types with neural networks.

Finally, the final method was applied to the Aachen (Germany) as an example and the results were compared with less granular reference data sets regarding number of inhabitants and electrical energy demand. It was shown that the total number of inhabitants could be determined with a deviation of 22% and the energy demand with a deviation of 8%.

# Inhaltsverzeichnis

<b>Inhaltsverzeichnis</b>	<b>v</b>
<b>Abbildungsverzeichnis</b>	<b>vi</b>
<b>Tabellenverzeichnis</b>	<b>vii</b>
<b>Abkürzungsverzeichnis</b>	<b>viii</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Verwandte Arbeiten . . . . .	2
1.3 Offene Herausforderungen . . . . .	3
1.4 Ziel und Aufbau . . . . .	3
<b>2 Analyse in den Grundlagen</b>	<b>4</b>
2.1 Energietechnik . . . . .	4
2.1.1 Energiebedarf . . . . .	4
2.1.2 Gebäudetypen . . . . .	4
2.1.3 Energetische Stadtraumtypen . . . . .	5
2.2 Datenquellen . . . . .	7
2.2.1 Georäumliche Daten . . . . .	7
2.2.2 OpenStreetMap . . . . .	8
2.2.3 ALKIS . . . . .	9
2.2.4 Georeferenzierte Senkrechtbilder . . . . .	10
2.2.5 Zensus . . . . .	13
2.3 Maschinelles Lernen . . . . .	13
2.3.1 Machine-Learning Verfahrenstypen . . . . .	13
2.3.2 Bildverarbeitung mit Neuronalen Netzen . . . . .	14
2.3.3 Decision Tree und RandomForest . . . . .	17
2.3.4 K-Means Clustering . . . . .	17
2.3.5 Metriken . . . . .	18
<b>3 Modellierung</b>	<b>21</b>
3.1 Segmentierung der Gebäudepolygone . . . . .	21
3.1.1 Trainingsvorbereitung . . . . .	22
3.1.2 Trainingsdurchführung mit RasterVision . . . . .	24
3.1.3 Erstellung der Vorhersagen . . . . .	26
3.1.4 Speicherung der Ergebnisse in der Datenbank . . . . .	28
3.2 Bestimmung des Energiebedarfes . . . . .	30
3.2.1 Vorbereitung . . . . .	31

---

3.2.2	Zuordnung von Gebäudetypen mit RandomForest . . . . .	32
3.2.3	Zuordnung von Stadtraumtypen mit K-Means Clustering . . . . .	34
3.2.4	Zuordnung eines Energiebedarfes . . . . .	35
<b>4</b>	<b>Darstellung und Diskussion der Ergebnisse</b>	<b>37</b>
4.1	Exemplarische Untersuchungen der Gebäudesegmentierung . . . . .	37
4.1.1	Gegenüberstellung der verwendeten Trainingsdatensätze . . . . .	37
4.1.2	Versuche zu den Trainingseinstellungen für RasterVision . . . . .	41
4.1.3	Versuche zu weiteren einflussnehmenden Eigenschaften aufs Training .	43
4.2	Exemplarische Untersuchungen zu RandomForest/K-Means . . . . .	47
4.2.1	Optimierung der Zuordnung von Gebäudetypen . . . . .	47
4.2.2	Untersuchungen zu den Features für das K-Means-Clustering . . . . .	49
4.3	Exemplarische Anwendung des Verfahrens . . . . .	49
<b>5</b>	<b>Zusammenfassung und Ausblick</b>	<b>53</b>
5.1	Zusammenfassung . . . . .	53
5.2	Ausblick . . . . .	54
	<b>Literaturverzeichnis</b>	<b>55</b>
<b>A</b>	<b>Tabellen</b>	<b>60</b>
<b>B</b>	<b>Bilder</b>	<b>65</b>
<b>C</b>	<b>Exemplarischer Verfahrensablauf</b>	<b>75</b>
<b>D</b>	<b>Voruntersuchungen zu RandomForest</b>	<b>79</b>
<b>E</b>	<b>Voruntersuchungen zum K-Means-Clustering</b>	<b>82</b>

# Abbildungsverzeichnis

2.1	Gebäudefunktionen in Alkis im Vergleich . . . . .	10
2.2	Bodenauflösungen im Vergleich . . . . .	11
2.3	Architektur von Convolutional Neural Networks . . . . .	15
2.4	Vergleich der Architekturen CNN und FCN . . . . .	16
2.5	Skizze eines exemplarischen Decision Trees . . . . .	17
2.6	Darstellung einer binären Konfusionsmatrix . . . . .	19
3.1	Überblick über die Hauptschritte der Methodik . . . . .	21
3.2	Verfahrensschema der Gebäudesegmentierung . . . . .	22
3.3	Exemplarisches Feature-Label-Paar . . . . .	25
3.4	Exemplarisches Feature-Label-Paar mit Vorhersage-Label . . . . .	26
3.5	Vergleich der Polygone vor und nach Postprocessing . . . . .	27
3.6	Gerasterte Polygone mit 1 Meter als Pixelsize . . . . .	28
3.7	Auszug der wichtigsten Tabellen der Datenbankstruktur des ersten Teils . . . . .	29
3.8	Verfahrensschema der Bestimmung des Energiebedarfs . . . . .	30
3.9	Beispiel des Prozesses zur Vorklassifizierung . . . . .	31
3.10	Erzeugung der Trainingsdaten für RandomForest . . . . .	32
3.11	Auszug der ALKIS und generierten Baublöcke in Aachen . . . . .	34
4.1	Vergleich von ALKIS und OSM bzgl. aller erfassten Gebäude in der Region Köln . . . . .	38
4.2	Berechnete Scores bei der Erkennung von allen Gebäude für ALKIS und OSM . . . . .	39
4.3	Vergleich von ALKIS und OSM bzgl. Wohngebäude in der Region Köln . . . . .	40
4.4	Ergebnisse der Erkennung von Wohngebäuden für ALKIS und OSM . . . . .	40
4.5	Untersuchungen zur Lernrate und Batchgröße . . . . .	41
4.6	Untersuchungen zur Epochs-Anzahl mit Fehlerkosten und Score . . . . .	42
4.7	Untersuchungen mit der ResNet101 Architektur . . . . .	43
4.8	Einfluss unterschiedlicher Siedlungstypen im Training und in der Anwendung . . . . .	44
4.9	Einfluss unterschiedlicher Bodenauflösungen im Training und in der Anwendung . . . . .	45
4.10	Einfluss unterschiedlicher WebMapServices im Training und in der Anwendung . . . . .	46
4.11	Ausgewertete Metriken für die Erkennung von Bauweisen durch spezialisierte neuronale Netze . . . . .	47
4.12	Konfusionsmatrizen für RandomForest-Klassifikation in 4 Bauweisen . . . . .	49
4.13	Vergleich der Einwohnerverteilungen zwischen Zensus und eigenen Berechnungen . . . . .	50
4.14	Berechneter Energieverbrauch in den Aachener PLZ-Gebieten im Vergleich . . . . .	52
B.1	Die Siedlungsräume aus Strukturperspektive von oben betrachtet . . . . .	66
B.2	Gegenüberstellung von TrueDOP mit DOP, historischen Aufnahmen und verschiedenen Bodenauflösungen des NRW Web Map Services . . . . .	67
B.3	Durch ein neuronales Netz erkannte Straßenzüge in der Region Aachen . . . . .	67

---

B.4	Manuell erstellter Decision Tree auf Grundlage der Kennzahlen für die energetischen Stadtraumtypen . . . . .	68
B.5	Darstellung der Vorgehensweise beim Auswählen der Area of Interests für die exemplarischen Untersuchungen . . . . .	68
B.6	Weitere eingeführte Regionen für WMS Untersuchungen bei selber Bodenauf- lösung . . . . .	69
B.7	Vergleichsmatrix von Web Map Services mit verschiedener Bodenauflösungen	69
B.8	Vollständige Korrelationsmatrix aller 34 Features für RandomForest . . . . .	70
B.9	Korrelationsmatrix der stark korrelierenden Features für RandomForest . . . . .	71
B.10	Vollständige Korrelationsmatrix aller 26 Features für K-Means . . . . .	72
B.11	Korrelationsmatrix mit stark korrelierenden Features für K-Means . . . . .	73
B.12	Zensus-Verteilung und eigene Vorhersage der Einwohnerzahlen im 100 m × 100 m Gitter in Aachen . . . . .	74
B.13	Postleitzahlgebiete im Stadtbezirk Aachen . . . . .	74
C.1	Abbildungen zu Aachen als Analyseregion und Köln als Trainingsregion . . . . .	75
C.2	Vorhergesagte Gesamtgebäude und Wohngebäude . . . . .	77
C.3	Vorklassifizierung der Aachener Wohngebäude . . . . .	77
C.4	Vorhergesagten Wohngebäude in Aachen wurde einer Bauweise zugewiesen .	78
C.5	Anwendung des K-Means-Clustering auf die künstlich generierten Baublöcke .	78
D.1	Auszüge aus der Korrelationsmatrix für RandomForest . . . . .	79
D.2	Konfusionsmatrizen für Random-Forest-Klassifikation in 5 Bauweisen . . . . .	81

# Tabellenverzeichnis

2.1	Deutsche Gebäudetypologie nach dem Institut Wohnen und Umwelt . . . . .	5
2.2	Auflistung der zehn energetischen Stadtraumtypen für Siedlungsräume mit Zuteilung des Nutzungstypes . . . . .	6
3.1	Mit PostGIS berechnete Features auf Grundlage der Gebäudepolygone . . . . .	33
3.2	Erste Merkmale zur Einteilung mit K-Means in die energetischen Stadtraumtypen	35
A.1	Kennwerttabelle für die energetischen Stadtraumtypen 1 bis 10 . . . . .	61
A.2	Gebäudebauweisen mit Beschreibung nach ALKIS Katalog . . . . .	62
A.3	Vergleich der Auflösung, Aktualität und Kosten des DOP-Angebotes der Bundesländer . . . . .	63
A.4	Hauptteil der an die Overpass API gestellten Query-Anfragen . . . . .	63
A.5	Geeignete Features für das K-Means Clustering . . . . .	64

# Abkürzungsverzeichnis

<b>ALKIS</b>	Amtliches Liegenschaftskatasterinformationssystem
<b>AOI</b>	Area of Interest
<b>CNN</b>	Convolutional Neural Network
<b>DH</b>	Doppelhaushälfte
<b>DOP</b>	digitale Orthofotografie
<b>EE</b>	Erneuerbare Energien
<b>EFH</b>	Einfamilienhaus
<b>EST</b>	Energetischer Stadtraumtyp
<b>GHD</b>	Gewerbe/Handel/Dienstleistungen
<b>GIS</b>	Geoinformationssystem
<b>GSD</b>	Ground Sampling Distance
<b>MCC</b>	Matthews Correlation Coefficient
<b>MFH</b>	Mehrfamilienhaus
<b>ML</b>	Machine Learning
<b>NDVI</b>	Normalized Difference Vegetation Index
<b>NIR</b>	Nah-Infrarot
<b>NN</b>	Neuronales Netz
<b>NRW</b>	Nordrhein-Westfalen
<b>OOI</b>	Object of Interest
<b>OSM</b>	OpenStreetMaps
<b>OAPI</b>	Overpass API
<b>RH</b>	Reihenhaus/gereihtes Haus
<b>TABULA</b>	Typology Approach for Building Stock Energy Assessment
<b>PV</b>	Photovoltaik
<b>WFS</b>	Web Feature Service
<b>WMS</b>	Web Map Service

# 1 Einleitung

## 1.1 Motivation

Im Jahr 2015 wurde durch das Pariser Klimaabkommen vereinbart, den globalen Temperaturanstieg auf 1,5 °C zu begrenzen [IPC21]. Um die dafür notwendige CO<sub>2</sub>-Emissionsreduktion zu erreichen, trat das Erneuerbare-Energien-Gesetz bereits im Jahr 2000 in Kraft, welches den Ausbau der Stromerzeugung aus erneuerbaren Energien antreiben sollte [MAT17].

Seitdem begann ein bis jetzt anhaltender Transformationsprozess im elektrischen Energiesystem. So wandelte sich die Nutzung von einigen wenigen Großkraftwerken hin zu vielen verteilten Erneuerbaren-Energien-Anlagen [BAG19]. Bis spätestens 2038 soll in Deutschland die Verstromung von Kohle sogar vollständig beendet werden [BMU21]. Die Möglichkeiten für regenerative Großanlagen wie Wasserkraftwerke sind Deutschland weitgehend ausgeschöpft und somit werden vermehrt kleinere Photovoltaik- oder Windparks gebaut [EIL12]. Diese Anlagen stellen in ihrem lokalen Netzbereich überschüssigen Strom bereit. Dem gegenüber steht der zunehmende Ausbau von Lasten wie der Elektromobilität und Wärmepumpen. [KLE20]. Diese neuen Stromerzeuger und Lasten bieten dabei unterschiedliche Herausforderungen und Möglichkeiten im Bereich der Energietechnik, wie z.B. den Verteilnetzausbau oder die Entwicklung von Quartieren, die im Rahmen der Forschung untersucht werden.

Zur Beantwortung der verschiedenen Forschungsfragen auf Verteilnetzebene werden Informationen über den Energiebedarf der Regionen mit hoher Granularität benötigt. Meistens ist der Energiebedarf jedoch nicht öffentlich zugänglich (z. B. wegen des Datenschutzes), sodass er durch geeignete Modelle bestimmt werden muss. Geografische Gebäudeinformationen lassen sich zwar von öffentlichen Quellen, wie dem Kartendienst OpenStreetMap, ohne Einschränkungen entnehmen, um mit geeigneten Modellen den Gebäuden Energiebedarfe zuzuordnen, jedoch ist die Beschaffenheit der Daten oftmals nicht ausreichend. Einige Gebäude sind entweder nicht verzeichnet oder falschen Energiesektoren zugeordnet [HEC13].

Eine neue potentielle Datenquelle für die Ermittlung der Gebäude bieten Satelliten- oder Luftbildaufnahmen. Mit Hilfe des maschinellen Lernens können auf diesen Bildern Objekte erkannt und extrahiert werden, sodass die fehlenden Daten automatisiert ergänzt werden können. Daher stellt sich die Frage nach einem Modell, das bildgestützt Gebäude identifizieren und ihnen Energiebedarfe zuordnen kann.

## 1.2 Verwandte Arbeiten

Im Folgenden werden zunächst Arbeiten über die Thematik der Modellierung des Energiebedarfes in kleineren Regionen vorgestellt. Die letztere Arbeit greift zusätzlich das Thema der Bildverarbeitung auf.

In der Arbeit von Howard et al. wurde ein gebäudescharfes Energiebedarfs-Modell auf Grundlage der Gebäudenutzung (z. B. Wohnung, Büro oder Lager) entwickelt. Zur Vorhersage wurde ein robustes lineares Regressionsmodell genutzt. Die Stadtverwaltung von New York City bietet eine georeferenzierte Datenbank zur Gebäudenutzung an, sodass das Modell auf die entsprechende Stadt angewendet werden konnte. [HOW12]

Ein Verfahren zur realitätsnahen Generierung von regionalen Gewerbe- und Haushaltslasten wurde von Augustyniak-Gaspers entwickelt. Dabei wurden frei verfügbare geografische und sozioökonomische Datenquellen verwendet. Aus OpenStreetMap wurden geografische Informationen, wie Gebäudeobjekte und Flächennutzungen, entnommen. Die sozioökonomischen Daten wurden aus Unternehmensregistern, Standardlastprofilen und Adressdatensätzen extrahiert. Nach einer Bereinigung der geografischen Daten, wurde versucht diese den sozioökonomischen Daten zuzuordnen. Da die Datenbestände nicht deckungsgleich sind, wurde statistisch versucht diese anzugleichen. [AUG20]

Schmid beschäftigte sich in seiner Dissertation ebenfalls mit der Modellierung der regionalen und zeitlich aufgelösten Energienachfrage. Die gewonnenen Informationen wurden innerhalb einer Rasterdarstellung visualisiert. Ein Rasterfeld fasst dabei den Energiebedarf der zugrunde liegenden Region zusammen. Als Eingangsdaten für das Modell nutzt Schmid ähnliche Arten von Quellen wie Augustyniak-Gaspers. Neben Kartendaten wurden Satellitendaten verwendet, um Prognosen für den Wärmebedarf von Städten anhand der Landflächennutzung abzuleiten. [SCH19a]

Mit einer genaueren Klassifizierung der Landflächennutzung setzte sich Jia in seiner Arbeit auseinander. Auf Basis von Orthofotos und LiDar-Sensordaten, wurde ein Ansatz für eine objekt-basierte Klassifizierung präsentiert. Lidar-Daten wurden genutzt um ein digitales Höhenmodell der Landschaft zu konstruieren, welches dann Strukturen wie Gebäude erkennen konnte. Die Nahinfrarot-Informationen der Orthofotos konnten für eine Spektralanalyse genutzt werden, um Vegetation und Wasser von den Gebäuden zu trennen. Die Verwendung von LiDar und Orthofoto-Informationen innerhalb eines Decision-Tree-Modells erzielten genaue Ergebnisse bei der Klassifikation einer exemplarischen Landschaft. [JIA15]

### 1.3 Offene Herausforderungen

Für Studien und Planungen im Bereich der Energietechnik wird eine gebäudescharfe Datenauflösung benötigt [SCH16]. Öffentliche Kartendienste können dem nicht ausreichend gerecht werden. So vermutet Augustyniak-Gaspers die Ursache einer auftretenden Diskrepanz zwischen amtlichen und kartografischen Gebäudedaten in OpenStreetMap [AUG20]. Auch Schmid bestätigt, dass zu den Nachteilen von OpenStreetMap die fehlende Validierung durch öffentliche Instanzen zählt. Je nach Region ist der Detaillierungsgrad sehr volatil [SCH19a]. Zusammenfassend zeigen die vorgestellten Arbeiten auf, dass es an einer allgemeinen Grundlage der Gebäudedaten fehlt. Diese wird bisher ausschließlich durch amtliche Quellen substituiert, was eine Adaption der Modelle in andere Regionen erschwert. Somit kann die Methode von Howard et al. nur in Städten angewendet werden, die eine öffentliche Gebäudenutzungsdatenbank verwalten. Außerdem ist eine Darstellung in größeren Gebietseinheiten (z. B. im Rasterformat) nicht zielführend. Neben der exakten Position ist auch der Umriss der Grundfläche der Gebäude zur Bestimmung des Energiebedarfes relevant [SCH19a]. Eine bildgestützte Analyse könnte dieses Problem lösen. Da eine Unmenge an auszuwertenden Daten zur Verfügung steht, ist eine händische Auswertung der Bildinformationen zu zeitaufwändig und zu teuer [JIA15]. Jia stellte dafür zwar einen Ansatz vor, der es ermöglicht Gebäude in Bilddaten algorithmisch zu finden, jedoch müsse der Decision-Tree nach eigenen Angaben für andere Regionen manuell angepasst werden.

### 1.4 Ziel und Aufbau

Das Ziel dieser Arbeit ist die Entwicklung eines Modells zur bildgestützten Identifikation von Gebäuden zur regionalen Bestimmung des Energiebedarfs. Dabei setzt es sich aus zwei Teilen zusammen. Im ersten Teil soll im Allgemeinen gezeigt werden, wie sich anhand von öffentlichen Luftbildern Gebäudeinformationen über bisher unbekanntere Regionen erlangen lassen. Der zweite Teil baut auf den zuvor erhobenen Daten auf. Die identifizierten Gebäudeinformationen sollen nun nach ihrem voraussichtlichen Energiebedarf klassifiziert werden. Mit dem zu entwickelnden Verfahren soll der Energiebedarf von ganzen Regionen abgeschätzt werden können.

Es wird zunächst mit einer Einführung in die Energietechnik begonnen. Anschließend werden die zu verwendenden Datenquellen im Kontext der Geoinformatik vorgestellt. Da die zu verwendenden Bilderkennungsverfahren aus dem Fachgebiet des maschinellen Lernens kommen, werden hierzu einige Grundlagen vorgestellt. Im Hauptteil wird dann die entwickelte und implementierte Methodik vorgestellt. Im letzten Kapitel des Hauptteils werden exemplarische Untersuchungen der Zwischenschritte der Methodik durchgeführt und die Ergebnisse diskutiert. Außerdem wird die entwickelte Methodik exemplarisch angewendet und das Ergebnis mit einem vergleichbaren Datensatz verglichen. Abschließend wird in Kapitel 6 die vorliegende Arbeit zusammengefasst und ein Ausblick auf zukünftige Forschungsmöglichkeiten geboten.

## 2 Analyse in den Grundlagen

In diesem Kapitel werden die Grundlagen der Arbeit vorgestellt und analysiert. Zunächst werden Grundbegriffe aus dem Bereich der Energietechnik in Abschnitt 2.1 definiert. Anschließend werden die in der Implementierung genutzten Datenquellen hinsichtlich ihres Datenbestandes und der Struktur beschrieben. Im letzten Abschnitt werden neben den genutzten Machine-Learning-Verfahren geeignete Metriken vorgestellt, mit denen die Ergebnisse im Diskussions- teil evaluiert werden können.

### 2.1 Energietechnik

Das Ziel dieser Arbeit ist es am Ende einen elektrischen Energiebedarf zu bestimmen. Daher wird zunächst der Begriff Energiebedarf definiert. Der Energiebedarf ist von unterschiedlichen Parametern abhängig, wie dem Gebäudetyp oder dem Stadtraumtyp. Deshalb werden beide Typologien anschließend in eigenen Abschnitten eingeführt.

#### 2.1.1 Energiebedarf

Der Bedarf an elektrischer Energie unterscheidet sich zwischen den Energiesektoren erheblich. Eine Einteilung der Sektoren findet in private Haushalte, Gewerbe/Handel/Dienstleistungen (GHD), Industrie und Verkehr statt. Unter dem GHD Sektor sind Bedarfe im Handwerk, Einzelhandel und von Dienstleistungsanbietern zusammengefasst. Der Sektor Industrie umfasst Industriebetriebe, die z. B. Produktionsstätte beinhaltet. Der Energiebedarf in den Sektoren GHD und Industrie hängt stark von dem Wirtschaftszweig, in dem das Unternehmen tätig ist, ab. Im Rahmen dieser Arbeit wird sich auf den elektrischen Energiebedarf im Sektor der privaten Haushalte beschränkt. [ENE19]

Der elektrische Energiebedarf in Haushalten umfasst die Nutzung von elektrischen Geräten zum Kochen, Waschen oder zur Unterhaltung. Die für den Betrieb eines elektrischen Gerätes benötigte Leistung wird als Wirkleistung bezeichnet und in Watt angegeben. Die Angabe der genutzten elektrischen Leistung ist in der Einheit Kilowattstunden (kWh) üblich [EMF21].

#### 2.1.2 Gebäudetypen

Im Energiesektor der privaten Haushalte hängt der Energiebedarf eng mit dem Gebäudetyp zusammen. Ein zu untersuchendes Wohngebäude muss deshalb als Erstes dem entsprechenden Typen zugeordnet werden, um eine Aussage über die energetischen Kennwerte treffen zu können. Im Rahmen des Projektes *Typology Approach for Building Stock Energy Assessment* (TABULA) wurden für 20 europäische Länder Wohngebäudetypologien aufgestellt. Dabei wurde versucht auf Basis der typologischen Kriterien synthetische Durchschnittsgebäude zu

erstellen, wobei dieses für Untertypen repräsentativ sein soll. Anhand dieser Durchschnittsgebäude kann dann eine Klassifizierung der energetischen Qualitäten eines ausgewählten Wohngebäudes stattfinden.

Ein Ergebnis war dabei, dass die Energieeffizienz der Gebäude maßgeblich durch die Gebäudegeometrie, Gebäudegröße, Nachbarsituation und das Baualter bestimmt wird. Das Baualter hat einen großen Einfluss auf den Bedarf, da sich in jeder Bauepoche eine eigene Konstruktionsweise durchgesetzt hat. Dabei unterscheiden sich Merkmale wie die Fenstergröße oder das genutzte Baumaterial. Neuere Epochen sind von Verordnungen geprägt, wie der Wärmeschutz- oder Energieeinsparverordnung. [LOG15]

Die deutsche Gebäudetypologie basiert auf dem Vorschlag des *Institutes Wohnen und Umwelt* und ist in Tabelle 2.1 aufgeführt [IWU20].

Kürzel	Name	Wohnungssituation
EFH	Einfamilienhaus	1 bis 2 Wohnungen
RH	gereihtes Haus	1 bis 2 Wohnungen als Doppelhaus
MFH	Mehrfamilienhaus	3 bis 12 Wohnungen
GMH	großes Mehrfamilienhaus	13 oder mehr Wohnungen
HH	Hochhaus	-

**Tabelle 2.1:** Deutsche Gebäudetypologie nach dem Institut Wohnen und Umwelt [IWU20]

Da in dieser Arbeit der elektrische Nutzenergiebedarf ermittelt werden soll, sind die einhergehenden Eigenschaften zum Baujahr nicht weiter relevant. Diese beeinflussen vielmehr den Heizenergiebedarf [IWU20]. Der elektrische Nutzenergiebedarf hingegen ist von der Anzahl der Personen abhängig. Anhand der beschriebenen Wohnsituation in der Tabelle lässt sich erkennen, dass die Einwohnerzahl abhängig von der Gebäudeklasse ist.

### 2.1.3 Energetische Stadtraumtypen

Im vorherigen Unterabschnitt wurde erklärt, dass der Energiebedarf von einzelnen Gebäudetypen abhängig sein kann. Zusätzlich ist es sinnvoll die Gebäude nicht nur isoliert zu betrachten, sondern ihre Einbettung in der urbanen Umgebung im Verbund mit anderen Gebäuden. Dafür werden verschiedene Grundstücke einer Stadt in bauähnliche Blöcke zusammengefasst. Ein Block wird durch öffentliche Straßenräume oder Wasserwege begrenzt. Außerdem lassen die Blöcke sich in voneinander abgrenzbare Typen anhand ihrer Eigenschaften klassifizieren. Eine Systematik zur Typologisierung dieser Blöcke wurde im Rahmen des Forschungsprojektes UrbanReNet erstellt [URB12]. Den Blöcken konnte in Folgeprojekten je ein energetisches Potenzial und ein Energiebedarf zugeordnet werden, sodass die Blöcke als energetische Stadtraumtypen (EST) bezeichnet werden. [HEG20]

Diese schlägt eine Einteilung in 25 energetische Stadtraumtypen und 23 energetische Einzelelemente vor. Bei den Einzelelementen handelt es sich um Sonderbauten, wie Kirchen, Krankenhäuser oder Hochschulen, die aufgrund ihrer spezifischen Nutzung und der baulichen Komplexität jeweils eine eigene energetische Betrachtung benötigen. Da solche Einzelelemente

seltener im Stadtbild vertreten sind, ist der elektrische Energiebedarf dieser vernachlässigbar, sodass die Einzelemente im Folgenden nicht weiter betrachtet werden.

Die 25 EST sind wiederum in zwei Klassen aufgeteilt: Siedlungsräume und Freiräume. Siedlungsräume bezeichnen bebaute Gebiete, die entweder bewohnt sind oder die zur gewerblichen Nutzung verwendet werden. Als Freiräume sind nicht bebaute Flächen innerhalb eines Stadtgebietes zu verstehen. Dies sind beispielsweise begrünte Anlagen, Verkehrsflächen und Flächen mit landwirtschaftlicher Nutzung. Solche Flächentypen tragen einen vernachlässigbar kleinen Anteil zum Energiebedarf einer Region bei, sodass im Folgenden ausschließlich auf die Siedlungsräume eingegangen wird. [HEG20]

Insgesamt gibt es zehn baulich abgrenzbare Siedlungsräume, die in Tabelle 2.2 aufgelistet sind. Im Anhang sind zusätzlich schematische Zeichnungen zu finden, wie diese Siedlungstypen in der Strukturperspektive von oben aussehen (siehe Abbildung B.1).

EST	Bezeichnung	Nutzungstyp
1	kleineteilige, freistehende Wohnbebauung	Überwiegende Wohnnutzung
2	Reihenhausbebauung	
3	Zeilenbebauung	
4	Großmaßstäbliche Wohnbebauung	
5	Blockrandbebauung	
6	dörfliche Bebauung	Überwiegende Mischnutzung
7	historische Altstadtbebauung	
8	Innenstadtbebauung	
9	Geschäfts-, Büro- und Verwaltungsgebiet	Überwiegend gewerbliche Nutzung
10	Gewerbegebiet	

**Tabelle 2.2:** Auflistung der zehn EST für Siedlungsräume mit Zuteilung des Nutzungstypes [HEG20]

Für das jeweilige Bauland eines EST-Blocks wurden Kennwerte zur Bebauungs- und Freiraumstruktur veröffentlicht. Anhand dieser charakteristischen Kennzahlen lassen sich zuvor definierte Blöcke einem der zehn energetischen Stadtraumtypen zuordnen. Dabei sind die Kennwerte stets auf 1 ha Nettobauland normiert worden. Als Nettobauland werden bebaute oder zur Bebauung vorhergesehene Flächen bezeichnet [ERL21].

Die energetischen Stadtraumtypes sind durch folgende Eigenschaften gekennzeichnet: Blockgröße, durchschnittliche Gebäudegrundfläche innerhalb des Blocks, Anzahl der Gebäude je Block, Anzahl der Gebäude je Hektar, Grundflächenanteil und der Freiflächenanteil. Anhand dieser Eigenschaften lassen sich die EST bestimmen.

Ist der EST-Typ eines Baublockes einmal bestimmt, können Aussagen über die Wohneinheiten, Einwohnerzahl und die Wohnfläche pro Hektar Nettobauland getroffen werden. Die Einwohnerzahl kann außerdem über die Dachfläche der Wohnbebauungen errechnet werden. Zusätzlich wird für die EST 1 bis 8 der Energiebedarf an Strom in  $\frac{\text{MWh}}{\text{ha}}$  angegeben. Für die EST 9 und 10 kann keine Aussage getroffen werden, da sich der Bedarf an Energie für gewerbliche Gebiete nicht überregional vereinheitlichen lässt [HEG20].

Der Energiebedarf lässt sich auf einen Bedarf pro Gebäudequadratmeter wie folgt umrechnen:

$$\text{Energiebedarf pro Gebäude} = 1000 \cdot \frac{\text{Energiebedarf des EST}}{\text{Anzahl der Gebäude}} \left[ \frac{\text{MWh/ha}}{1/\text{ha}} = \text{kWh} \right] \quad (2.1)$$

$$\text{Energiebedarf pro Quadratmeter} = \frac{\text{Energiebedarf pro Gebäude}}{\text{durchschnittliche Gebäudegrundfläche}} \left[ \frac{\text{kWh}}{\text{m}^2} \right] \quad (2.2)$$

Mit dieser Angabe lässt sich allein durch die Grundfläche eines Gebäudes eine Schätzung des Energiebedarfes vornehmen. Beispielsweise liegt der Energiebedarf eines Reihenhauses bei 49 kWh pro Quadratmeter im Durchschnitt. Eine Auflistung mit den genannten Kennzahlen und den Umrechnungen für den Energiebedarf ist in Tabelle A.1 im Anhang für alle Siedlungsräume zu finden. [HEG20]

## 2.2 Datenquellen

Zur Bestimmung des Energiebedarfs einer zu analysierenden Region – auch Area of Interest (AOI) genannt – werden unterschiedlichste Eingangsdaten aus den Bereichen der Geografie und Verwaltung benötigt. Die im Folgenden vorgestellten Datenquellen stellen Informationen bereit, die stets mit einer geografischen Position verknüpft sind. Aus diesem Grund werden im ersten Abschnitt Begrifflichkeiten aus dem Bereich der Geowissenschaften definiert. Anschließend werden mit OpenStreetMap und ALKIS Datenbanken vorgestellt, die den Gebäudebestand einer AOI (mehr oder weniger gut ausgeprägt) abbilden können. Im Abschnitt Digitale Orthofotografien wird eine spezielle Art von Luftaufnahmen erklärt und über welche Dienste die Aufnahmen bereitgestellt werden. Als letztes wird auf den Zensus eingegangen, der eine Datengrundlage zur Evaluierung der Ergebnisse anbietet.

### 2.2.1 Georäumliche Daten

Als georäumliche Daten (Geodaten) werden Informationen bezeichnet, die sich in einem georäumlichen Koordinatensystem einordnen lassen. Unter dem Begriff Georeferenzierung wird der Prozess zur Zuweisung der Referenzinformation an eben diese Daten verstanden. Die Georeferenzierung bei gerasterten Bildgrafiken geschieht durch die Zuordnung von Bildpunkten zu Koordinaten. Bei vektoriellen Formaten, wie Polygonen, muss ein fest bestimmter Nullpunkt einer Koordinate zugeordnet werden. [SPE20]

Zur konkreten Zuweisung von Koordinaten muss ein passendes geodätisches Bezugssystem (engl. coordinate reference system, kurz CRS) gewählt werden. Das Bezugssystem setzt sich aus einem Festpunktfeld und dem Koordinatensystem zusammen. Unter Festpunkten werden eindeutig identifizierbare Vermessungspunkte bezeichnet. Das Koordinatensystem kann entweder kartesisch oder ellipsoidisch sein. Bei kartesischen Koordinatensystemen stehen die jeweiligen Achsen orthogonal zueinander, sodass die euklidischen Gesetze bezüglich Abstands- und Flächenmessungen gelten.

In einem ellipsoidischen Koordinatensystem beziehen sich die Koordinaten auf ein Referenzellipsoid. Dadurch eignet sich letzteres Koordinatensystem, um die gekrümmte Erdoberfläche annähern zu können. Die Koordinaten eines Punktes werden in dem Kontext als Breiten- und Längengrad bezeichnet. Ein weltweit gebräuchliches ellipsoidisches CRS ist das *World Geodetic System 1984* (WGS84).

Anhand des europäischen EPSG-Verzeichnisses wird eine vereinheitlichte Übersicht mit Schlüsselkennungen der verfügbaren CRS angeboten. Die dort hinterlegten CRS entsprechen der anerkannten Norm des Open Geospatial Consortium. Das WGS84 trägt in diesem Verzeichnis die Kennung EPSG:4326. Neben den globalen Bezugssystemen gibt es lokal angepasste Projektionen. Diese Projektionen nutzen das kartesische Koordinatensystem, welche eine metrische und winkeltreue Abbildung der Flächen in der lokalen Region ermöglichen. In Deutschland werden oftmals die Gauß-Krüger-Projektionen (EPSG:31466) verwendet. [GEO20]

Um Geodaten auf Rechnern verarbeiten zu können, werden Geoinformationssysteme (GIS) verwendet. Mit der PostGIS Erweiterung für PostgreSQL ist beispielsweise die Verwaltung von Geodaten innerhalb einer objektrelationalen Datenbank möglich. PostGIS führt neben den bekannten Datentypen sogenannte Geometrietypen ein. Dies sind z. B. Punkte (POINT), einfache Linien (LINE) oder einzelne Polygone (POLYGON). Genauso wie es für die ursprünglichen Datentypen Indexstrukturen zur effizienten Suche gibt, lassen sich räumliche Indizes für die Geometrietypen anlegen. PostGIS nutzt dafür intern zu Indexierung die Datenstruktur der R-Bäume.

Um auf den neuen Datentypen Operationen ausführen zu können, werden durch PostGIS neue Funktionen angelegt. Zum Beispiel gibt es Funktionen um Distanzen (ST\_DISTANCE) oder Flächen (ST\_AREA) zu berechnen. Die interne Tabelle *spatial\_ref\_sys* umfasst über 3000 EPSG-Codes und Umrechnungsparameter, sodass das Übersetzen von Geodaten in ein anderes Bezugssystem mit der Funktion ST\_TRANSFORM einfach durchzuführen ist.

Eine grafische Visualisierung der in PostGIS hinterlegten Geodaten kann wiederum in einer GIS-Software stattfinden. So ist QGIS eine mit PostGIS kompatible freie Open-Source GIS-Software zur Datendarstellung. [POS21]

### 2.2.2 OpenStreetMap

Der Kartendienst OpenStreetMap (OSM) ist ein 2004 gegründetes Open-Source-Projekt, welches das Ziel verfolgt eine frei zugängliche Weltkarte anzubieten. Unter der Open-Database-Lizenz wird sichergestellt, dass die hinterlegten Daten stets öffentlich bleiben und sich jeder beteiligen kann. Somit können Freiwillige beliebig Wege, Gebäude und anderen Ortsinformationen in der OSM-Datenbank ergänzen. [OSM21d]

Die grundlegenden Elemente, auf denen alle Daten in OSM aufbauen, sind Punkte, Linien und Relationen. Während OSM zur Darstellung der Daten auf Pseudo-Mercator (EPSG:3857) als CRS setzt, wird zur internen Datenhaltung EPSG:4326 genutzt, sodass die Position eines Punktes durch Breitengrad und Längengrad bestimmt ist. Linien haben ebenfalls einen

geografischen Bezug und können Straßen, Flüsse oder Grenzen modellieren. Relationen sind Gruppen von Objekten und sind keinem Ort geografisch zugeordnet. Durch eine Relation werden Beziehungen unter den teilnehmenden Objekten dargestellt.

Den beschriebenen Grundelementen lassen sich Attribute zuweisen. Ein Attribut besteht dabei aus einem Schlüssel-Wert Paar und kann zusätzliche Informationen über ein Objekt bereitstellen. Dabei entspricht eine Attributzuweisung dem Format `Schlüssel=Wert`. Beispielsweise lässt sich mit `elevation=2962` die Höhe eines Punktes auf 2962 m über dem Meeresspiegel festlegen. Die Kombination der Grundelemente erlaubt es dann reale Objekte, wie Straßen, Wälder und Gebäude zu beschreiben [OSM21b].

Für die Beschreibung von Gebäuden wird ein besonderes Element benötigt: die Fläche. Eine Fläche wird in OSM über geschlossene Linien (Anfangs- und Endpunkt sind identisch) definiert. Diese spezielle Art der geometrischen Figur ist als Polygon bekannt. Komplexere Flächen werden als Multipolygon bezeichnet und werden über Relationen definiert.

Der Zugriff auf die Geoinformationen in OSM geschieht mit Hilfe der Overpass API (OAPI). Durch die OAPI wird es ermöglicht nach bestimmten Attributen und Objekttypen innerhalb einer quadratischen Region zu suchen. Die Koordinaten der quadratischen Region werden als Bounding Box bezeichnet. Eine Abfrage an die OAPI wird in der eigenen Overpass Query Language formuliert. Beispielsweise lassen sich über eine Abfrage alle Flächen mit dem Attribut `building=yes` selektieren. Dies entspricht der Abfrage nach Gebäude-Polygonen innerhalb einer Bounding Box. Antworten werden im GeoJSON-Format zurückgegeben. Die GeoJSON-Spezifikation ist eine Erweiterung der JSON-Notation, welche ein maschinenlesbares Datenformat zum Austausch von Informationen ist. GeoJSON erweitert JSON um Geometrietypen, wie es bei PostGIS und PostgreSQL der Fall ist [TIF21]. Mit dem Software-Paket `osm2postgis` lassen sich an OSM gestellte Anfragen direkt in eine PostGIS Datenbank übertragen. [OSM21c]

### 2.2.3 ALKIS

Das Akronym ALKIS steht für *Amtliches Liegenschaftskatasterinformationssystem* und ist seit dem 01. Dezember 2015 ein deutschlandweit eingeführtes System zur Verwaltung der Liegenschaftskataster. Der Datenbestand von ALKIS umfasst über eine halbe Millionen Gebäude, Flurstücke und diverse Grenzmarkierungen. Im Gegensatz zu OpenStreetMap geschieht das Einpflegen von neuen Gebäudedaten nicht durch Freiwillige, sondern durch die deutschen Vermessungsverwaltungen. Es lassen sich zudem nicht beliebige Daten aus ALKIS auslesen. Stattdessen wurde ein länderübergreifender Grunddatenbestand definiert, der als Mindestinhalt den Nutzern zur Verfügung steht. [ALK21][BER21]

Die Daten im ALKIS Modell werden über Objektbereiche kategorisiert und voneinander getrennt verwaltet. Zum Beispiel wird durch den Objektbereich `AX:Baublock` Zugang auf die als Nettobauland ausgedehnten Bereiche einer Gemeinde gewährt. Ein weiterer nützlicher Bestandteil der öffentlich verfügbaren Grunddaten ist die Objektart `AX:Gebäude`. Damit ist es möglich den Gebäudebestand in ganz Deutschland abzufragen. Dies ist zwar auch in OSM möglich, jedoch umfassen die in ALKIS hinterlegten Gebäude weitere für den Energiebedarf

hilfreiche Attribute, wie die Gebädefunktion, Gebäudebauweise oder das Merkmal Hochhaus. Der Wertebereich der Gebädefunktion unterscheidet 233 mögliche Ausprägungen. Dies reicht von Altersheimen und Bürogebäuden hin zu Studentenwohnheimen oder Wasserwerken. Die verschiedenen Ausprägungen lassen sich angelehnt an die Energiesektoren auf fünf Kategorien reduzieren: Wohngebäude, GHD, Industrie, Sonstige und kein Energiebedarf. Die Kategorie Sonstige umfasst beispielsweise Verwaltungsgebäude, Sportanlagen und schulische oder religiöse Einrichtungen. Unter kein Energiebedarf sind Gebäude wie Garagen oder Gartenhäuser als Gebädefunktion aufgeführt. In Abbildung 2.1 sind die Gebädefunktionen der Aachener Innenstadt farblich in der ausführlichen und reduzierten Variante visualisiert worden. [ALK18] Für die in ALKIS hinterlegten Gebäudebauweisen (zwölf mögliche Ausprägungen) lässt sich



(a) Alle Gebädefunktionen - 233 mögl. Werte

(b) Reduzierte Gebädefunktionen - 5 mögl. Werte

**Abbildung 2.1:** Abgebildet sind Gebäude und ihre zugeschriebenen Funktionsweisen (ausführlich in a und reduziert in b) im dem Bereich der Aachener Innenstadt [ALK21]

ebenfalls eine Vereinfachung vornehmen, sodass am Ende nur noch in fünf Bauweisen unterschieden wird. Die Bauweisen weisen parallelen zur TABULA Gebäudetypologie auf, da die Einteilung ebenfalls in Ein- und Mehrfamilienhäuser unterscheidet. So ist EST 1 von EFH und MFH geprägt, während in EST 5 die geschlossene Blockbauweise charakteristisch ist. Eine Übersicht der ALKIS Werte und im Kontext dieser Arbeit genutzten Bauweisen findet sich in Tabelle A.2. [ALK18]

Um letztendlich auf die Daten in ALKIS über eine API zugreifen zu können, wird ein Web Feature Service (WFS) angeboten. Dabei handelt es sich um eine HTTP-Schnittstelle, die die Objekte in Form von Vektordaten ausliefert. Über die Software QGIS lässt sich das spezielle Vektordatenformat dann wiederum in eine GeoJSON übersetzen. [WFS20]

## 2.2.4 Georeferenzierte Senkrechttbilder

Bei georeferenzierten Senkrechttbildern handelt es sich um Aufnahmen, die senkrecht zur Erdoberfläche geschossen und einer exakten geografischen Position zugeordnet wurden. Zunächst werden Eigenschaften wie die Auflösung und der Informationsgehalt solcher Aufnahmen definiert. Anschließend wird mit digitalen Orthofotos ein Subtyp von Senkrechttbildern vorgestellt.

## Auflösung und Informationsgehalt

Die Qualität der Aufnahmen lässt sich anhand der Bodenauflösung (GSD) feststellen. Die Bodenauflösung beschreibt, wie viele Meter ein Pixel in der Realität abbilden würde. Um kleinere Strukturen, wie Autos oder Bäume, auf solchen Bildern erkennen zu können, sollte die Bodenauflösung höher als  $1 \frac{\text{m}}{\text{pixel}}$  sein [AZA21]. Ein Vergleich von einer höheren gegenüber einer niedrigeren Bodenauflösungen findet sich in Abbildung 2.2.



(a) 10cm pro Pixel - DOP in NRW [WMS21a]      (b) 10m pro Pixel - Sentinel Satellitenbild [WMS21b]

**Abbildung 2.2:** Bodenauflösungen im Vergleich anhand eines Siedlungsgebietes mit dichter Bebauung. In (a) sind die Umrisse einzelner Häuser gut zu erkennen, während in (b) sich nur noch gröbere Strukturen wie Straßen und Häusern auseinander halten lassen.

Neben der Auflösung ist die Art der aufgezeichneten Information relevant. Dies können übliche RGB-Bilder oder Infrarot-Bilder sein. Je nach aufgezeichneter Wellenlänge kann sich der Informationsgehalt eines Bildes unterscheiden. Das besondere an den Wellenlängen ist, dass sie abhängig vom Material mit einer spezifischen Intensität reflektiert werden. Diese eindeutige Charakteristik wird als spektrale Signatur bezeichnet und ermöglicht es vice versa Rückschlüsse auf das Material zu ziehen. Auf der Grundlage von Nah-Infrarot (NIR) und des roten Farbkanals lässt sich mit dem Normalized Difference Vegetation Index (NDVI) bereits das Material einzelner Pixel klassifizieren. Gesunde Vegetation absorbiert im roten Farbspektrum (ROT) und reflektiert im NIR-Spektrum die Lichtwellen. Der NDVI berechnet sich wie folgt:

$$\text{NDVI} := \frac{\rho_{\text{NIR}} - \rho_{\text{ROT}}}{\rho_{\text{NIR}} + \rho_{\text{ROT}}}, \text{ wobei } \text{NDVI} \in [-1, 1] \quad (2.3)$$

NDVI-Werte im stark negativen Bereich deuten auf Wasser hin, nahe 0 handelt es sich um urbanisierte Gebiete und Werte im stark positiven Bereich sprechen für Vegetation [GIS21].

## Digitale Orthofotografien

Im Zusammenhang mit Senkrechtaufnahmen werden oftmals Satellitenaufnahmen assoziiert. Jedoch ist der Zugang zu diesem Bildmaterial sehr eingeschränkt und die Qualität der öffentlich verfügbaren Aufnahmen oberhalb von  $10 \frac{\text{m}}{\text{pixel}}$  [AZA21]. Eine Alternative zu Satellitenbildern bieten Luftbildaufnahmen von Flugzeugen. Unbearbeitete Luftbildaufnahmen würden aber eine falsche räumliche Zuordnung und abweichende Maßstäbe der realen Umgebung vermitteln. Das Verfahren der Orthorektifizierung (lat.: Berichtigung) nutzt bekannte Bodenkontrollpunkte zur geografischen Zuordnung und ein digitales Höhenmodell zur Entzerrung. Nach

dem Prozess der Rektifizierung wird eine hochauflösende digitale Orthofotografie (DOP) generiert, welche die Geometrie des aufgezeichneten Gebietes korrekt wiedergibt. Die digitale Eigenschaft ermöglicht es diese Fotos als Basiskarten in einem GIS oder für die weitere Bildverarbeitung zu nutzen. Neuere Fotografien sind vom Typ TrueDOP. Bei der Berechnung normaler DOP wird von höheren Gebäuden die seitliche Fassade erfasst, sodass die Gebäude gekippt erscheinen. Dieser Effekt ist bei TrueDOP Bildern nicht mehr vorhanden. Zusammenfassend sind die Vorteile von DOP eine günstige Anfertigung und Bodenauflösungen von unter  $1 \frac{\text{m}}{\text{pixel}}$ . [LEA97]

In Deutschland sind die Vermessungsverwaltungen der Bundesländer gesetzlich verpflichtet worden die aktuelle Topografie festzuhalten. Dies beinhaltet neben der Gebäudeverwaltung in ATKIS das Erstellen von DOP [ATK21]. Die DOP werden oft mit einer jährlichen Aktualisierung gepflegt, sodass hier ein weiterer Vorteil gegenüber freien Satellitenbildern besteht. Das Bundesamt für Kartografie und Geodäsie stellt auf der Plattform Geoportal<sup>1</sup> die Luftbilddaufnahmen von insgesamt zwölf Bundesländern zur Verfügung. Einige Bundesländer, wie Baden-Württemberg oder Niedersachsen, erheben für die Datennutzung eine Gebühr [DOP21a] [DOP21b]. Andere Bundesländer, wie Nordrhein-Westfalen (NRW), stellen die Bilder zur Nutzung frei [NRW21]. Die frei verfügbaren Aufnahmen sind entweder unter der Creative Commons 3 Lizenz (CC3) oder der Datenlizenz Deutschland Version 2.0 (dl-de/by-2-0) veröffentlicht worden. Unter beiden Lizenzen wird das teilen und bearbeiten unabhängig von der Nutzung uneingeschränkt erlaubt. Lediglich eine Namensnennung wird verlangt. Eine komplette Auflistung der Verfügbarkeit von DOP innerhalb Deutschlands findet sich im Anhang in Tabelle A.3. Neben den aktuellen Aufnahmen bieten viele Bundesländer Zugriff auf das historische Archiv an, sodass teilweise DOPs aus der Mitte des letzten Jahrhunderts zugänglich sind. Abschließend ist eine Gegenüberstellung von TrueDOP zu DOP, historischen Aufnahmen und höheren Bodenauflösungen mit Abbildung B.2 angehängt.

Für die Darstellung von DOP wird das GeoTiff-Format bevorzugt genutzt und von den Bundesländern bereitgestellt. Tiff-Dateien (Tagged Image File Format) sind verlustfreie Rastergrafiken, die zusätzliche Bildinformationen in Form von Tags hinterlegen können. Bei GeoTiff-Dateien wurden Geoinformationen in Form von georeferenzierten Tags hinzugefügt. Damit lassen sich einzelne Bilddateien ohne Hilfe anderer Quellen eigenständig einem Ort zuordnen. [KOS20]

Um auf DOP zugreifen zu können wird ein Web Map Service (WMS) bereitgestellt. Der WMS stellt die Daten genau wie ein Web Feature Service über ein HTTP-Interface zur Verfügung. Im Gegensatz zum WFS werden nicht Vektordaten zurückgegeben, sondern gerasterte Bilddaten einer Karte. Zum Abrufen der Bilddaten wird die Funktion GetMap in den Request-Parametern verwendet. Daneben wird eine Bounding Box, das CRS, der Layer (z. B. RGB oder NIR), das Bild-Ausgabeformat, und die Höhe/Breite der auszugebenen Bildkachel benötigt [WMS21c][WMS20]. Als CRS lässt sich von allen Bundesländern entweder EPSG:3857 oder EPSG:4326 nutzen.

---

<sup>1</sup><https://www.geoportal.de/>

### 2.2.5 Zensus

Bei dem Zensus handelt es sich um eine in Deutschland strichprobenartig durchgeführte Volkszählung. Die Erhebungen sind regelmäßig in einem 10 Jahresabstand geplant. Zuletzt wurde der Zensus deutschlandweit am 09. Mai 2011 durchgeführt. Die nächste Datenerhebung wurde auf Grund der Corona-Krise erst wieder im Jahr 2022 geplant [ZEN21]. Zusätzlich zu den Befragungen durch den Zensus werden verschiedene Melderegister zusammengeführt und Ungenauigkeiten statistisch bereinigt, sodass eine deutschlandweit umfassende Datenbank zu sozialen und ökonomischen Fragen jeweils zum Zensus-Stichtag erstellt wird. Dieser Datensatz beinhaltet z. B. Zahlen zur Bevölkerung, Erwerbstätigkeit und Wohnsituation in ganz Deutschland. Die Daten stehen unter der Datenlizenz Deutschland 2.0 unter Namensnennung frei zur Verfügung. [ZEN11]

Im Übrigen wurden alle Informationen mit Hilfe eines Gittersystems georeferenziert. Innerhalb eines 1x1 Kilometer oder 100x100 Meter Rasters werden die gesammelten Werte aggregiert, um unter anderem dem Datenschutz gerecht zu werden. [SCH19a]

## 2.3 Maschinelles Lernen

Methodiken aus dem Bereich des Maschinellen Lernens (ML) werden immer dann eingesetzt, wenn die Komplexität der Aufgabe nicht mit der klassischen Programmierung zu lösen ist. Zuerst wird auf die 3 Lernstrategien des ML eingegangen. Anschließend werden mit neuronalen Netzen im Bereich der Bildverarbeitung, RandomForest und K-Means-Clustering konkrete Implementierung der Lernstrategien vorgestellt. Zuletzt werden einige Metriken definiert, mit denen sich die ML-Ergebnisse evaluieren lassen.

### 2.3.1 Machine-Learning Verfahrenstypen

Eine Aufgabe, die sich nicht mit der klassischen Programmierung lösen lässt, ist beispielsweise das Segmentieren von Objekten auf Bildern. Das Verfahren der Segmentierung wird als die Zuordnung von einzelnen Pixeln zu Objekten bezeichnet. Es handelt sich bei den Objekten um zusammenhängende Pixelbereiche, die z. B. Bäume, Häuser oder Autos darstellen können [SAB13]. In der klassischen Programmierung würde ein Programmierer einen Anforderungskatalog, der das Input/Output-Verhalten eines Programms festlegt, erhalten und seine anschließende Aufgabe wäre es das zugehörige Modell als Programmcode zu schreiben. Das Problem ist jedoch, dass sich die komplexeren Objekte auf jedem Bild unterscheiden können. Schließlich gibt es keine einfachen Regeln nach denen sich die Erkennung eines komplexen Objektes beschreiben lässt.

Im Kontext von ML werden die Input-Werte als Features und die Output-Werte als Labels bezeichnet. Der ML-Ansatz ersetzt den Programmierer durch eine ML-Strategie, die selbstständig ein passendes Modell zu den Features und Labels finden soll. Insgesamt gibt es drei Arten von ML-Lernstrategien: Supervised, Unsupervised und Reinforcement Learning. [NGU18]

Die Strategie des **Supervised Learning** (zu dt.: überwachtes Lernen) setzt auf ein induktives Lernen mit Beispielen. Ein Algorithmus lernt anhand von verschiedenen Trainingsdaten ein oder mehrere Muster kennen. Die Trainingsdaten setzen sich aus Features und dem dazu erwarteten Label zusammen. Dem Algorithmus wird damit Feedback gegeben, ob es die Features dem richtigen Label zuordnen konnte, oder ob es sein erlerntes Modell nochmal anpassen muss. Es wird dabei in zwei verschiedenen Anwendungen unterschieden: Klassifikation und Regression. Im Anwendungsgebiet der Klassifikation wird versucht verschiedene Klassen aufgrund ihrer Merkmale voneinander zu trennen. Die Anwendung der Regression zielt darauf ab Vorhersagen zu geben. Neuronale Netze (NN) und Support Vector Machines sind zwei bekannte Lernalgorithmen, die beide Anwendungsgebiete beherrschen. [SCH19b]

Das **Unsupervised Learning** (zu dt.: unüberwachtes Lernen) erzeugt ein Modell ohne explizites Training. Auf der Grundlage von Input-Werten versucht der Algorithmus eine interne Repräsentation der Daten zu finden. Die interne Darstellung muss durch den Menschen anschließend im Gegensatz zum Supervised Learning manuell klassifiziert werden. Ein typisches Anwendungsgebiet für das Unsupervised Learning ist das Clustering. Lernalgorithmen für das Unsupervised Learning sind K-Means oder Hierachical Clustering. [SCH19b]

Beim **Reinforcement Learning** (zu dt.: verstärkendes Lernen) interagiert ein Agent (der Algorithmus) mit seiner Umgebung, da die Trainingsdaten selbstständig generiert werden müssen. Der Agent erhält stetig positives (Belohnung) oder negatives (Bestrafung) Feedback von seiner Umgebung abhängig von seiner gewählten Aktion. Das Ziel des Agenten ist es eine Strategie zu entwickeln mit der er seine Belohnung maximiert. Demnach verfolgt das Reinforcement Learning Konzept einen spieletheoretischen Ansatz. Ein Algorithmus ist beispielsweise das Q-Lernen. [ERT16]

### 2.3.2 Bildverarbeitung mit Neuronalen Netzen

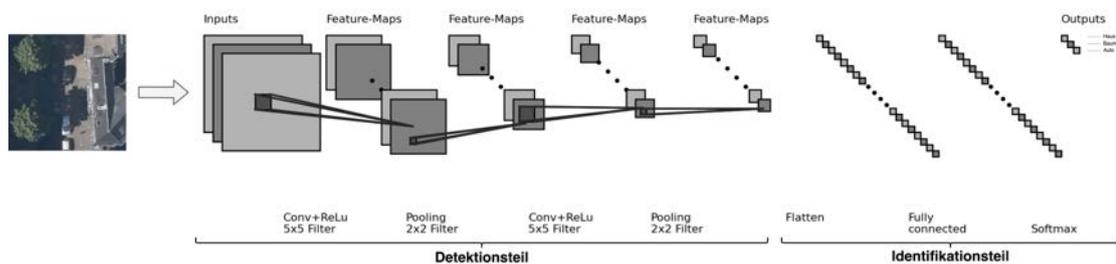
Die Segmentierung von Objekten auf Bildern fällt in den Bereich der Bildverarbeitung. Für die Segmentierung in der Bildverarbeitung wird das Verfahren des Supervised-Learnings verwendet. Als Lernalgorithmus werden dafür oftmals NN genutzt. Hauptbestandteil der NN sind die Neuronen. Jedes Neuron erwartet einen Eingabewert, verarbeitet es mit einer internen Funktion und generiert einen Ausgabewert. Neuronen lassen sich je nach genutzter Funktion einem Layer zuordnen, der eine bestimmte Aufgabe haben kann. Eine Anordnung von Layern wird durch Architekturen beschrieben.

Die Neuronen sind untereinander mit Kanten verbunden, die Gewichte tragen. Die Anpassung der Kantengewichte passiert während eines Trainings und ermöglicht es dem Netzwerk Eigenschaften zu erlernen. Um die Kantengewichte anpassen zu können, erwartet das NN Features und Labels. Ein Feature wird in Form eines Eingangsvektors an die ersten Neuronen des Netzwerks übergeben. Das NN trifft eigene Vorhersagen auf dem Feature und gleicht die Ergebnisse mit dem passenden Label ab. Je nach Abweichung zum Label wird anhand einer internen Strategie berechnet, welche Kantengewichte, wie stark angepasst werden. [SCH19b]

Im Kontext der Bildverarbeitung werden die Bilder als Features bezeichnet. Bildinformationen lassen sich als zweidimensionale Matrix modellieren, wobei ein Feld die Intensität des Pixels

wiedergibt. Bei RGB-Bildern wird demnach je Farbkanal eine eigene Bildmatrix benötigt. Im Allgemeinen ist der Eingangsvektor eines NN eindimensional, da einzelne Features durch Skalare beschrieben werden können. Es wäre zwar möglich die zweidimensionalen Bildinformationen in einen eindimensionalen Vektor zu transformieren, jedoch würden dann wertvolle Nachbarschaftsinformationen verloren gehen.

Damit die Nachbarschaftsinformationen erhalten bleiben, wird für die Bildverarbeitung mit ML häufig eine spezielle Architektur der NN verwendet: das Convolutional Neural Network (CNN). Die allgemeine Struktur der Architektur ist der Abbildung 2.3 zu entnehmen. Üblicherweise setzt sich ein CNN aus dem Detektions- und dem anschließenden Identifikationsteil zusammen. [SCH19b]



**Abbildung 2.3:** Architektur von Convolutional Neural Networks (DOP von [WMS21a])

Der **Detektionsteil** setzt sich aus mehreren sogenannter Convolutional-Layern und Pooling-Layern zusammen. Der Eingangsvektor für die Convolutional-Layers ist das Bild bzw. bei einem RGB-Bild die drei jeweiligen Kanäle. Im Convolutional-Layer selber nimmt jedes Neuron nur einen kleinen Ausschnitt, der als Filter bezeichnet wird, des Bildes wahr. Zudem soll jede Ebene an Neuronen, die als Feature-Map bezeichnet wird, lernen auf ein anderes Muster im Filter zu reagieren. Auf den Convolutional-Layer folgt ein Pooling-Layer, der die Dimensionalität verringern soll. Dies reduziert folglich die Anzahl der zu erlernenden Gewichte für die Neuronen der nachfolgenden Feature-Maps und lässt das Lernen schneller konvergieren. Oftmals wird dafür das Max-Pooling genutzt. Innerhalb eines Filters wird dafür der maximale Wert bestimmt und dieser dann als repräsentative Information an die nächste Schicht weitergeleitet.

Nach dem Detektionsteil, der Strukturen extrahieren soll, werden die Informationen durch den **Identifikationsteil** interpretiert. Für den Identifikationsteil wird ein mehrlagiges Fully-connected-NN genutzt. Dieses erwartet einen eindimensionalen Eingangsvektor. Ein Flatten-Layer wandelt deshalb die dreidimensionale Ausgangsmatrix des letzten Pooling-Layers in das benötigte Format um. Am Ende des Fully-connected-NN wird ein Softmax-Layer auf die Output-Neuronen angewendet. Dieser bestimmt mit welcher Wahrscheinlichkeit ein Bild zu einer Klasse (z. B. Haus, Auto, Baum) gehört. [SCH19b]

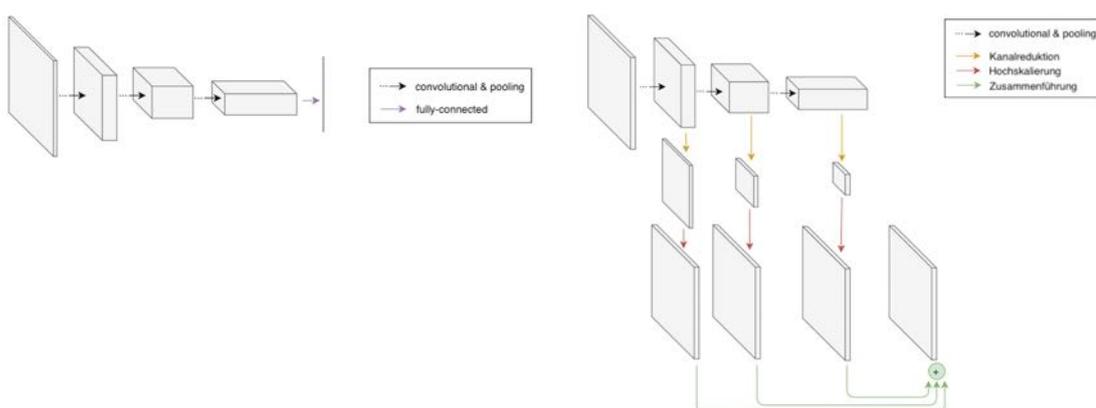
### Bildererkennung mit RasterVision

RasterVision ist ein Open-Source-Framework [FIS21] der gemeinnützigen Organisation Azaeva für Deep Learning mit Satelliten- und Luftbildern. Es kann auch für nicht georeferenzierte

Senkrechtaufnahmen, wie geneigte Drohnenaufnahmen, genutzt werden. Beispielhafte Anwendungen von RasterVision sind die Segmentierung von Häusern oder Fahrzeugen.

RasterVision hat keine standardmäßigen Datenquellen integriert und setzt sich das Ziel ein flexibles Framework für beliebige Bilderkennungs-Anwendungen zu bieten. Bei der Verwendung von RasterVision wird dem Entwickler ein Workflow zur Verfügung gestellt. Der einheitliche Workflow soll es den Nutzern erleichtern schnelle wiederholbare Experimente durchzuführen und diese untereinander vergleichbar zu machen. Um das Berechnen des geeignetsten Modells zu beschleunigen, ist es möglich mehrere Experimente simultan laufen zu lassen. Während des Trainings werden dann schlechter konvergierende Modelle vorzeitig ausgeschlossen. Außerdem können als Eingangsdaten mehrere Bildlayer auf einmal verwendet werden. Damit ist es z. B. möglich RGB- und NIR-Bilder gleichzeitig für das Training zu nutzen. [EMA18]

Als Architektur nutzt RasterVision eine angepasste Variante des CNNs: Das Fully Convolutional Network (FCN). Beim normalen CNN entsteht am Ende nur ein eindimensionaler Vektor. Dieser gibt je Label die Wahrscheinlichkeit an, ob dieses im gesamten Bildbereich vorgekommen ist. Jedoch soll diese Beurteilung für möglichst kleine Bereiche funktionieren, sodass am Ende folgende Aussage getroffen werden kann: Der Pixel ist Bestandteil eines Objektes oder der Pixel ist Teil des Hintergrundes (aber nicht beides). Nur so ist das Ergebnis für eine Segmentierung von Gebäuden brauchbar. Das FCN verzichtet daher auf den Fully-Connected-Layer, der bei einem CNN im Identifikationsteil vorhanden ist. Stattdessen werden aus den Zwischenlayern die Informationen aus den Feature-Maps entnommen und im Ersten Schritt auf die Anzahl der möglichen Label-Ausprägungen (Kanalreduktion) reduziert. Anschließend werden die kleineren Bildmatrizen auf die ursprünglich Bilddimension interpoliert. Im letzten Schritt werden alle gesammelten Informationen zusammengeführt, sodass eine Label-Zuordnung für jeden Pixel stattfinden kann. Eine schematische Skizze beider Architekturen ist der Abbildung 2.4 zu entnehmen. [FIS17]



(a) Vereinfachte Darstellung eines CNN

(b) Vereinfachte Darstellung eines FCN

**Abbildung 2.4:** Vergleich der Architekturen CNN und FCN zur Bilderkennung [FIS17]

### 2.3.3 Decision Tree und RandomForest

Bei Decision Trees (zu dt. Entscheidungsbaum) handelt es sich um ein Supervised-Learning-Verfahren, welches für die Klassifizierung von Datensätzen genutzt werden kann. Es wird ein Modell trainiert, das mit Hilfe einfacher Entscheidungsregeln den Eingangsdaten die Label möglichst korrekt zuteilen kann. Vorteile von Entscheidungsbäumen sind die einfache Interpretation (z. B. durch Visualisierung), eine geringe Datenvorbereitung (Preprocessing) und das simultane Einordnen in mehrere Klassen. Ein Beispiel eines Entscheidungsbaumes mit drei Klassen und drei Entscheidungsregeln in Form von binären Features  $x_1, x_2, x_3$  ist in Abbildung 2.5 gegeben. [DTS21]

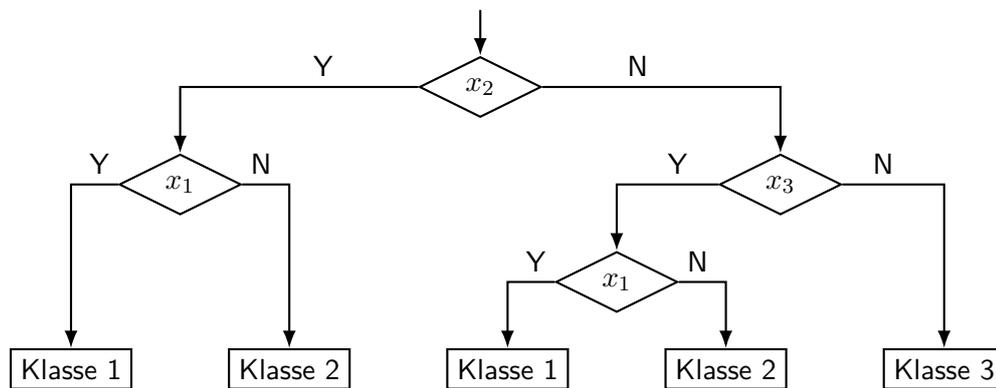


Abbildung 2.5: Skizze eines exemplarischen Decision Trees

Jedoch neigen einzelne Decision Trees dazu nur auf bekannten Daten gute Vorhersagen zu treffen (Overfitting), sodass Methoden zur Regularisierung angewendet werden müssen. Solche Methoden können das Festlegen der Mindestanzahl an Verzweigungen oder der Anzahl der Blätter im Entscheidungsbaum sein. Das Finden solcher idealen Regularisierungsparameter kann aufwendig sein. Eine Alternative zum einfachen Decision Tree stellt der RandomForest-Klassifikator dar, der weniger anfällig für die zuvor genannten Nachteile ist. Die RandomForest-Methode teilt die initiale Datenmenge in viele kleinere Mengen auf. Zusätzlich wird jeweils eine zufällige Teilmenge der Features genutzt. Auf jede dieser Mengen wird ein Entscheidungsbaum berechnet. Die Ergebnisse der unterschiedlichen Decision Trees werden am Ende mit gleicher Gewichtung zusammen gefasst, sodass ein „RandomForest“ entsteht. [NGU18]

### 2.3.4 K-Means Clustering

Bei K-Means handelt es sich um ein Unsupervised-Learning-Verfahren, das ungelabelte Daten in verschiedene Gruppen – auch Cluster genannt – einteilen soll. Dafür sollte ungefähr bekannt sein, wie viele Cluster es gibt. Innerhalb eines Clusters haben die Datenpunkte homogene Eigenschaften. Der Algorithmus teilt eine Menge  $N$  mit Werten  $x$  in  $k$  disjunkte Cluster  $C$  ein, wobei jeder Cluster anhand des Mittelwertes  $\mu_j$  beschrieben wird.

Zur Berechnung versucht der Algorithmus  $k$ -Mittelpunkte in die Cluster-Zentren zu setzen. Die Zentren werden im Laufe mehrerer Iterationen so verschoben, dass sie die kleinste Abweichung in Form der totalen Quadratsumme jeweils bilden [KME21]:

$$\sum_{i=0}^n \min_{\mu_j \in C} (\|x_i - \mu_j\|^2) \quad (2.4)$$

### 2.3.5 Metriken

Mit Hilfe von Metriken lassen sich Scores berechnen, die eine Aussage darüber treffen können, wie gut sich ein trainiertes Modell verhält. Ein Score hat oft das Intervall  $\{x \in \mathbb{R} \mid 0 \leq x \leq 1\}$ , wobei  $x$  nahe 1 für ein gutes und nahe 0 für ein schlechtes abschneiden innerhalb einer Metrik spricht. Anhand von verschiedenen Scores lässt sich dann entscheiden, ob es sinnvoll ist das Modell auf weiteren Daten in einer praktischen Anwendung zu nutzen. Metriken geben zudem Aufschluss darüber, ob ein trainiertes Modell zu Over- oder Underfitting neigt. Bei Overfitting erreicht das Modell nur auf bereits bekannte Daten ein gutes Ergebnis und bei Unbekannten wird die Einordnung eher schlecht bewertet. Das Modell hat in dem Fall nicht gelernt zu abstrahieren. Beim Underfitting kann das Modell nicht die Trainingsdaten zuverlässig klassifizieren, sodass es überhaupt keine Eigenschaften gelernt hat [NGU18]. Im Folgenden werden Metriken zur Bewertung des Supervised Learnings vorgestellt.

#### Accuracy

Eine einfache Metrik stellt *accuracy* dar. Es wird das Verhältnis der Datenbeispiele mit richtig vorausgesagten Labels zu der Anzahl aller untersuchten Datenbeispiele gegenübergestellt.

$$accuracy = \frac{n(\text{richtig vorhergesagt})}{n(\text{alle})} \quad (2.5)$$

Eine *accuracy* von 1 würde beispielsweise aussagen, dass alle Daten richtig klassifiziert werden konnten. Jedoch bedeutet eine hohe *accuracy* nicht immer sofort, dass ein Modell besser klassifizieren kann, als ein anderes mit einer niedrigeren Bewertung. In der binären Klassifikation kann es durch eine ungleiche Verteilung der Label zwischen zwei Klassen zum sogenannten *Accuracy Paradoxon* kommen [ACC17]:

Angenommen ein Modell  $\alpha$  ordnet die Daten stets der Klasse 1 zu und der wirkliche Anteil der Daten in Klasse 1 im Verhältnis zu Klasse 2 betrage 99%. Dann wäre die  $accuracy(\alpha) = 0,99$  für das Modell  $\alpha$ , da 1%, nämlich die Klasse 2 Daten, falsch zugeordnet wurden. Es sei desweiteren ein Modell  $\beta$  in der Lage alle Klasse 2 Daten korrekt zuzuordnen, jedoch kann es nun nur noch 98% aus Klasse 1 richtig klassifizieren. Die *accuracy* von Modell  $\beta$  wäre  $accuracy(\beta) = 0,99 \cdot 0,98 + 0,1 \approx 0,97$ . Obwohl Modell 1 überhaupt keine Eigenschaften der Klasse 2 gelernt hat, erreicht es wegen der ungleichen Klassenverteilung einen besseren Score. Um solche Missverhältnisse besser einschätzen zu können, lassen sich die anschließend vorgestellten Metriken nutzen.

## Konfusionsmatrix

Bevor die weiteren Metriken eingeführt werden, soll die Darstellungsform einer Konfusionsmatrix kurz erläutert werden. Die Zeilen einer solchen Matrix beschreiben die Realität. In der Bilderkennung mit Luftbildaufnahmen werden diese als Ground Truth (zu dt. Bodenwirklichkeit) bezeichnet. Die Spalten beschreiben die vorhergesagten Daten. Im Falle einer binären Klassifikation mit den Labeln *positive* und *negative* lässt sich in vier Fällen unterscheiden:

- true positive (*TP*): Anzahl der positive Label, die korrekt zugeordnet wurden
- false positive (*FP*): Anzahl der positive Label, die falsch zugeordnet wurden
- true negative (*TN*): Anzahl der negative Label, die korrekt zugeordnet wurden
- false negative (*FN*): Anzahl der negative Label, die falsch zugeordnet wurden

In Abbildung 2.6 ist eine binäre Konfusionsmatrix dargestellt. Eine solche Matrix kann auch für mehr als zwei Label erstellt werden. Diese ist stets quadratisch, sodass eine Maximierung der Werte entlang der Hauptdiagonalen angestrebt wird. Es besteht die Möglichkeit die absoluten Werte innerhalb einer Matrix zu normalisieren, indem die Werte der Felder durch die jeweilige Zeilensumme  $P'$  oder  $N'$  geteilt wird. [NGU18]

		Vorhersage		
		p	n	
Realität	$p'$	true positive	false negative	$P'$
	$n'$	false positive	true negative	$N'$
		$\Sigma$	$P$	$N$

**Abbildung 2.6:** Darstellung einer binären Konfusionsmatrix

## Precision und Recall

Die Precision Metrik beschreibt die Präzision mit der die Werte innerhalb einer bestimmten Klasse getroffen wurden. Umso weniger falsche Zuordnungen es zu dieser Klasse gibt, umso höher ist die Präzision.

$$precision = \frac{TP}{TP + FP} \quad (2.6)$$

Die Recall Metrik beschreibt dem Anteil aller korrekt zugeordneten Daten zu einer Klasse. Der Wert sagt aus, wie gut ein Modell Daten wieder erkennt bzw. wie gut die Abdeckung ist.

$$recall = \frac{TP}{TP + FN} \quad (2.7)$$

Beide Metriken lassen sich jeweils auf die zu untersuchenden Label anwenden. Zwar wird es angestrebt beide Scores zu maximieren, jedoch kann je nach Anwendung eine Metrik bevorzugt behandelt werden. [POW20]

### $F_\beta$ -Score

Um *precision* und *recall* in ein Verhältnis zu setzen, wird der  $F_\beta$ -Score genutzt. Mit  $\beta \in ]0, 1[$  wird *precision* stärker gewichtet und mit  $\beta > 1$  liegt der Fokus mehr auf dem *recall*-Wert [SAS07]:

$$F_\beta = (1 + \beta^2) \cdot \frac{\textit{precision} \cdot \textit{recall}}{(\beta^2 \cdot \textit{precision}) + \textit{recall}} \quad (2.8)$$

Für  $\beta = 1$  wird das harmonische Mittel beider Metriken gebildet, sodass die Eigenschaften Präzision und Wiedererkennung gleich stark gewichtet werden:

$$F_1 = 2 \cdot \frac{\textit{precision} \cdot \textit{recall}}{\textit{precision} + \textit{recall}} \quad (2.9)$$

Der F-Score ist eine oft genutzte Metrik im Bereich des maschinellen Lernens, jedoch wird kritisiert, dass der Anteil der *true negatives* nicht einbezogen wird. [POW20]

### Matthews Korrelationskoeffizient

Matthews Korrelationskoeffizient ( $MCC$ ) lässt sich nutzen, um alle Ergebnisse einer binären Klassifikation ins Verhältnis zu setzen. Die Berechnung des  $MCC$  geschieht wie folgt:

$$MCC = \frac{(TP \cdot TN) - (FP \cdot FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (2.10)$$

mit  $MCC \in [-1, 1]$

Ein Score nahe 1 sagt aus, dass das Modell alle Vorhersagen korrekt trifft, bei  $-1$  würde das Modell stets das Gegenteil klassifizieren. Befindet sich der Score nahe 0, so gleichen die Vorhersagen einem Zufallsexperiment. Der  $MCC$  eignet sich, falls beide binären Klassen mit selber Priorität zu behandeln sind. Schließlich wird eine ungleiche Verteilung zwischen beiden Klassen berücksichtigt. [SHM19]

### 3 Modellierung

In diesem Kapitel werden die Modellierungsschritte beider Teile dieser Arbeit erläutert. Das Verfahren setzt sich aus zwei Teilen zusammen. Um den Energiebedarf einer Region ermitteln zu können, müssen zunächst die Gebäude innerhalb dieser identifiziert werden. Die Methodik zur Segmentierung der Gebäude auf Grundlage von Luftbildaufnahmen wird in Abschnitt 3.1 gezeigt. Daran kann anschließend der Energiebedarfes bestimmt werden, was in Abschnitt 3.2 erklärt wird. Einen Gesamtüberblick bietet Abbildung 3.1.



Abbildung 3.1: Überblick über die Hauptschritte des ersten und zweiten Teils der Methodik

#### 3.1 Segmentierung der Gebäudepolygone

Das Ziel der Segmentierung von Gebäudepolygonen ist es Gebäudegrundrisse auf Basis von Ortofotos bestimmen zu können. Eine detailliertere Übersicht der Schritte des ersten Teils bietet Abbildung 3.2. Als Erstes werden die benötigten Daten für das Training eines NN vorbereitet. Nach der Vorbereitung kann dann die Trainingsdurchführung mit Hilfe von RasterVision stattfinden. Nachdem ein Netzwerk trainiert wurde, wird es auf die zu analysierende Region angewendet. Zuletzt werden die bestimmten und nachbearbeiteten Gebäudepolygone in einer Datenbank hinterlegt. Es werden also vier größere Bearbeitungsschritte durchlaufen, die in den anschließenden Unterabschnitten ausführlicher beschrieben sind.

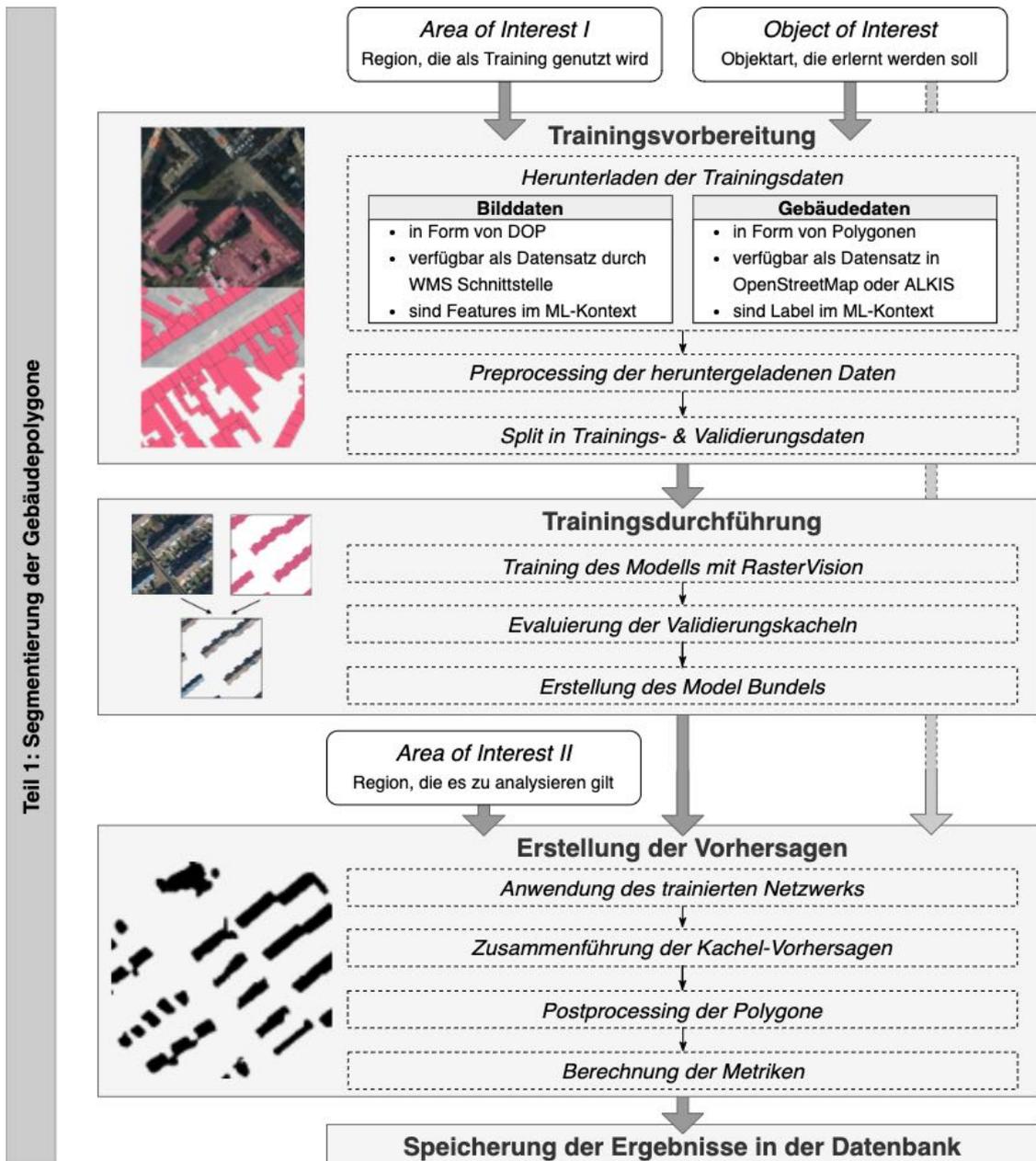


Abbildung 3.2: Verfahrensschema des ersten Teils der Modellierung

### 3.1.1 Trainingsvorbereitung

Als Eingangsparameter werden eine Area of Interest (AOI) und ein Object of Interest (OOI) benötigt. Innerhalb dieser AOI findet das Training statt, wobei das zu trainierende Modell die durch das OOI vorgegebene Objektart lernen soll. Die AOI wird in der Implementierung als GeoJSON-Datei im EPSG:4326-Code abgelegt. Mit Hilfe des gitHub Repository `boundary.now` können beispielsweise in OSM gespeicherte Verwaltungsgrenzen im benötigten Format heruntergeladen werden [YU17]. Als OOI kann beispielsweise auf alle Gebäudearten trainiert werden, oder ausschließlich auf bewohnte Gebäude.

Im initialen Schritt der Trainingsvorbereitung werden die Daten für das Training heruntergeladen. Das einzige Trainings-Feature im ersten Teil stellen die Bilddaten dar. Diese werden als

digitale Orthofotos von den entsprechenden WMS-Schnittstellen bezogen. Für die standardmäßige NRW-Schnittstelle stehen z. B. drei Layer zur Verfügung: RGB, NIR und CIR. Der CIR-Layer ist eine farbige Variante des NIR-Layers, der die Bilddaten normalerweise in Graustufen darstellt. Pixel mit einem NDVI-Wert nahe 1 werden auf CIR-Bildern rötlicher dargestellt, sodass auch für den menschlichen Betrachter die Vegetation im Fokus liegt. Als Bildformat werden GeoTiffs angefordert, da somit eine manuelle Georeferenzierung der Bilder ausbleibt. Soweit möglich wird als CRS die Mercator-Projektion (EPSG:4326 oder EPSG:3857) verwendet, da die Gebäudepolygone in OSM ebenfalls in diesem Bezugssystem liegen. Weitere variable Einstellungen zur Anfrage an einen WMS sind die Bildabmessungen und das Zoomlevel. Das Zoomlevel ist eine diskrete Angabe des Maßstabes in der WMS-Spezifikation mit dem Werteintervall  $[0 \dots 20] \subset \mathbb{N}$ . Die Weltkarte wird bei Wahl des Zoomlevels  $z$  in  $4^z$  Kacheln unterteilt. Für  $z = 0 \Rightarrow 4^0 = 1$  wird beispielsweise die gesamte Weltkarte als einzige Kachel zurückgegeben. Das Zoomlevel  $z = 17$  unterteilt die Welt in  $4^{17} = 17\,179\,869\,184$  Kacheln. Der Maßstab der Objekte auf einer solchen Kachel ist 1:4000. Dieser Maßstab eignet sich, um mehrere Gebäude im Rahmen eines Baublockes auf einer Kachel erfassen zu können [OSM21a]. Daher wird, falls nicht anders erwähnt, als Zoomlevel  $z = 17$  in allen später beschriebenen Durchführungen verwendet.

Um alle Bilddaten einer Trainingsregion herunterladen zu können, muss je Kachel ein Request an einen WMS gestellt werden, der für den entsprechenden Auszug der Region verantwortlich ist. Dafür wird zunächst das Polygon der AOI auf jene Kacheln abgebildet, die den Gesamtbereich abdecken. Jede Kachel entspricht den Koordinaten einer Bounding Box. Neben den zuvor erwähnten Konfigurationseinstellungen für die Services der Bundesländer, sind die Verwaltungsgrenzen zu diesen in der Software hinterlegt. Anhand der Koordinaten der Bounding Box, lässt sich bestimmen, welcher WMS zum Abrufen der Bilddaten angefragt werden muss. Die Geodaten der Bounding Box sind bei heruntergeladenen GeoTiffs in der Datei entsprechend kodiert.

Sollte der WMS keinen Download von GeoTiffs anbieten, wird ein PNG-Bild angefordert. Da es sich bei PNGs ebenfalls um Rastergrafiken handelt, ist eine Übersetzung in das GeoTiff-Dateiformat möglich. Dem konvertierten Bild wird dann die Bounding Box manuell mitgegeben. Falls das Bild nicht in der Mercator-Projektion ausgeliefert werden kann, muss das Bild anschließend in das erwähnte CRS konvertiert werden.

Bildkacheln, die auf der AOI-Grenze liegen, werden im letzten Schritt zugeschnitten. Durch das Zuschneiden wird sichergestellt, dass das Training nur im definierten Bereich stattfindet. Ohne Zuschneiden würde mehr Fläche als ursprünglich vorgegeben zum Training verwendet werden. Damit sind alle benötigten Bilddaten für das Training heruntergeladen und vorverarbeitet worden.

Neben den Features werden für das Training im Rahmen eines Supervised-Learnings mit NN Label benötigt. Im Kontext der Bilderkennung von Gebäuden sind die Label Gebäudepolygone. Die Label geben dem NN zu erkennen wie Gebäude überhaupt aussehen, indem sie dem NN mitteilen an welcher Position sich ein Gebäude im Bild befindet. Dadurch kann sich das Netzwerk im Training selbst prüfen, ob es die richtigen Eigenschaften eines Gebäudes lernt oder ob sich das Training in eine falsche Richtung entwickelt.

Als Datenquellen für die Label werden OpenStreetMap und ALKIS verwendet. Wird das Polygon über OSM bezogen, wird die OAPI verwendet. Diese benötigt ebenfalls eine Bounding Box innerhalb der die Geodaten bezogen werden sollen. Als Bounding Box werden die Koordinaten des kleinsten Rechtecks verwendet, das die Trainingsregion umschließt. Neben der Bounding Box wird die Query formuliert, die die gewünschten OOI abfragt. Das CRS der Polygone in der übermittelten GeoJSON-Datei wird von OSM in EPSG:4326 projiziert. Eine Übersicht über die Overpass QL Abfragen für die OOI ist in Tabelle A.4 hinterlegt.

Zuletzt müssen genauso wie für die Bilddaten alle Polygone außerhalb der Trainingsregion entfernt werden. Polygone die auf der Grenzlinie liegen werden entsprechend beschnitten.

Für Geodaten aus ALKIS müssen die OOI in der Verfahrensimplementierung manuell vorbereitet werden, da die WFS Schnittstelle keine eigene Query Language unterstützt. So lässt sich in QGIS über das Plugin *WFS Client 2.0* der ALKIS Gebäudebestand abfragen. Die in einer GIS-Software angebotenen Werkzeuge lassen sich dann nutzen, um die Polygone dem OOI-Typ entsprechend aufzubereiten. Anschließend lassen sich die gefilterten ALKIS Polygone als GeoJSON in der Mercator-Projektion exportieren. [WFS20]

Nachdem alle Bilder und Gebäudepolygone heruntergeladen und vorbereitet wurden, wird für jede Bildkachel eine Polygonkachel erstellt. Dafür wird über die GeoTiffs iteriert und die in den Metadaten hinterlegte Bounding Box extrahiert. Jede Bounding Box wird über die vollständig heruntergeladene Menge der Polygone gelegt. Anschließend wird diese jeweils auf den Bildbereich der Bildkachel zugeschnitten. Mit der Erstellung der Feature-Label-Paare ist das Daten-Preprocessing vollständig abgeschlossen.

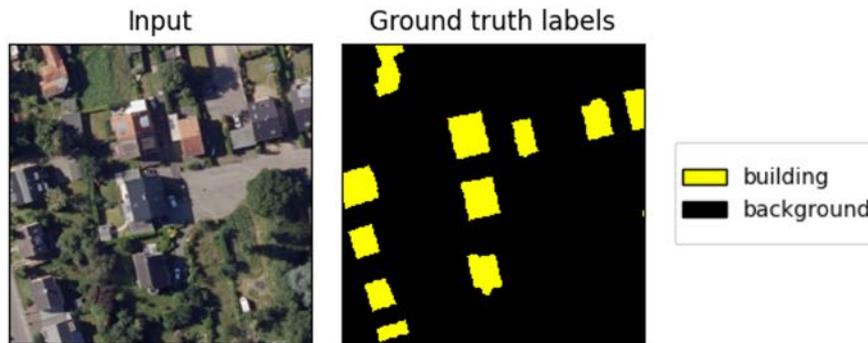
Als letztes wird die Gesamtmenge der Feature-Label-Paare in zwei disjunkte Teilmengen unterteilt. Eine Teilmenge dient dabei zum Training des NN und die weitere zur ersten Validierung innerhalb der Trainingsregion. Diese erste Validierung lässt vermuten, wie sinnvoll weitere Anwendungen mit diesem Modell sind. Das Verhältnis der Trainings zu Validierungsdaten beträgt standardmäßig 80% zu 20%, kann aber bei Bedarf anders gewählt werden.

### 3.1.2 Trainingsdurchführung mit RasterVision

Nachdem alle Daten im vorherigen Schritt vorbereitet wurden, kann das Training mit dem Framework RasterVision begonnen werden [FIS21]. RasterVision stellt vorimplementierte Pipeline-Konfigurationen für unterschiedliche ML-Anwendungen bereit. In diesem Fall wird die Pipeline zur semantische Segmentierung der Gebäudegrundrisse verwendet. Dafür wird dem Framework übergeben, welche Label es lernen soll: Gebäudepixel (im Folgenden `building`-Label genannt) und Hintergrundpixel (im Folgenden `background`-Label genannt). Da nur mit zwei Labeln trainiert wird, findet eine binäre Klassifikation statt. Neben dem Speicherort der digitalen Orthofotos wird dem Framework übergeben, welche Kanäle der Rastergrafiken zum Training verwendet werden sollen. Es können zum Training auch mehrere Bildlayer gleichzeitig genutzt werden. So können bei Bedarf mehrere Bildlayer (z. B. RGB und NIR) gleichzeitig als Feature für das Training geladen werden. [FIS21]

Anschließend werden die vorbereiteten Polygonkacheln abgerufen. Da es sich bei Polygonen

um Vektordaten handelt und bei den Bildern um Rasterdaten, müssen die Polygone noch gerastert werden. Das Übersetzen der Polygone in eine Rasterdatei übernimmt das Framework. Am Ende hat RasterVision, sowohl die Features, als auch Labels als gerasterte Daten zur Verfügung. Ein Beispiel für ein finales Feature-Label-Paar zum Training ist in Abbildung 3.3 abgebildet. Die in gelb markierten Pixel auf dem Ground Truth Rasterbild geben an, dass es sich beim Input an der selben Stelle um ein Gebäudepixel handelt.



**Abbildung 3.3:** Links das für das Training genutzte Feature [WMS21a], rechts das Label [OPE21]

Neben den oben genannten Parametern für die Trainingsvorbereitung, gibt es für die Trainingsdurchführung ebenfalls spezifische Einstellungen. Dies sind die Epochs, Batchgröße, Lernrate und welcher Backbone verwendet werden soll.

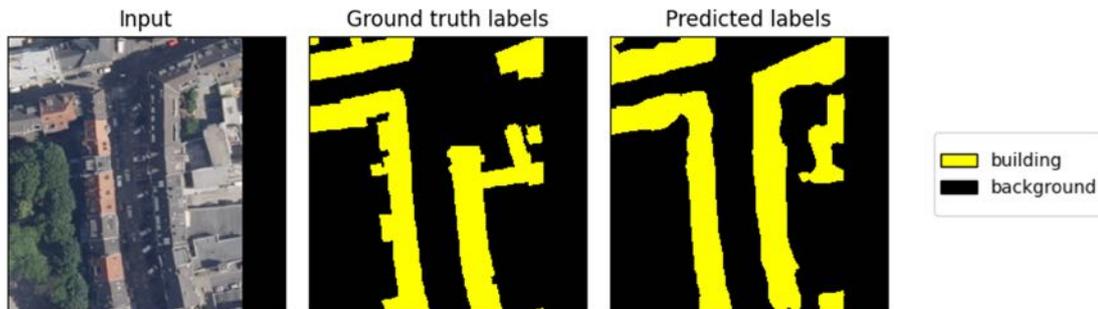
Die Epochs bezeichnen die Anzahl der Durchläufe durch die gesamte Datenmenge im Training. Beispielsweise würden mit  $x$  Bildern und einer Epochszahl von  $n$  dem NN im gesamten Training  $x \cdot n$  Bilder übergeben werden. Ist die Anzahl der vollständigen Iterationen zu klein gewählt, kann es sein, dass das NN nicht ausreichend lange lernen konnte. [BRO18]

Die Batchgröße beschreibt die Anzahl der Datensätze, die durch das Netzwerk ausgewertet werden sollen, bevor interne Parameter des Modells angepasst werden. Nach jedem Batch-Durchlauf prüft RasterVision mit Hilfe der Validierungsdaten, wie gut das aktuelle Modell abgeschnitten hat. Auf Grundlage eines Fehlerscores, werden dann Gewichte im Netzwerk des Modells verändert, was als eigentlicher Lernprozess bezeichnet werden kann. Wie stark die Gewichtungen angepasst werden, hängt von der Lernrate ab. Sie definiert, welche Schrittgröße zur Anpassung der Gewichtungen nach jeder Batch-Iteration genutzt werden soll. Ist die Schrittgröße zu klein eingestellt, dauert es sehr lange, bis ein Optimum gefunden wird, sodass mehr Epochs insgesamt notwendig sind. Bei einer zu großen Lernrate wird das Optimum eventuell nie gefunden. [BRO18]

Unter dem Begriff Backbone ist die Hintergrundarchitektur des Netzwerks zu verstehen. RasterVision erlaubt es entweder ResNet50 oder ResNet101 für das Training zu nutzen. Beide Netzwerke sind Instanzen der CNN-Architektur. Ersteres ist 50 Layer und letzteres 101 Layer tief. Für die Auswahl der spezifischen Einstellungen zur Trainingsdurchführung gibt es keine allgemeinen Empfehlungen. Diese müssen durch Experimente gefunden werden.

Nach dem Training des Modells wird von RasterVision eine Label-Vorhersage auf die zuvor definierten Validierungskacheln durchgeführt (siehe Abbildung 3.4). Dafür werden der *precision*, *recall* und  $F_1$ -Score für die Label *building* und *background* berechnet. Da-

durch kann der Anwender einschätzen, ob das trainierte Modell sich für die weitere Anwendung eignet. Außerdem lässt sich anhand der Abbildung erkennen, dass die Vorhersagen gereihte Gebäude nicht in ihre einzelnen Gebäudebestandteile trennen können. Daher sind im Folgenden unter vorhergesagten Gebäuden stets Gebäudeverbände zu verstehen.



**Abbildung 3.4:** links und mitte bilden das Feature-Label-Paar [WMS21a][ALK21], rechts die Label-Vorhersagen

Im letzten Schritt der Trainingsdurchführung wird von RasterVision ein Model-Bundle bereitgestellt. In diesem ist zum einen das trainierte NN enthalten, welches für weitere Vorhersagen genutzt werden kann. So ist es möglich dieses Bundle als Grundlage für weitere Trainings zu nutzen, wodurch das NN weiter verbessert werden kann.

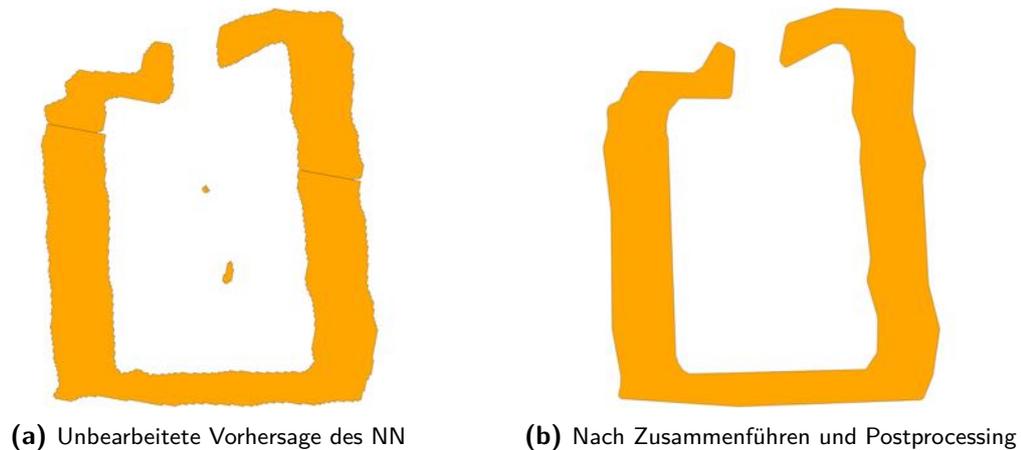
### 3.1.3 Erstellung der Vorhersagen

Nachdem das Netzwerk trainiert wurde und die ersten Auswertungen der Metriken eine sinnvolle Anwendung versprechen, lässt sich das Modell auf eine weitere AOI anwenden. Bei der zweiten AOI handelt es sich um die Region deren Energiebedarf in Erfahrung gebracht werden soll. Zur Vorhersage der Gebäudepolygone in der neuen Region werden die ersten zwei Teilschritte der Trainingsvorbereitung wiederholt (siehe Abbildung 3.2).

Auf den vorbereiteten Daten kann das zuvor erstellte Model-Bundle angewendet werden. Dafür wird jede Bildkachel dem trainierten NN übergeben, welches eine GeoJSON-Datei generiert, in der die vorhergesagten Gebäudeumrisse als Polygone kodiert sind. Der Inhalt der Datei wird in eine PostGIS Datenbank geladen, um die folgenden Verarbeitungsschritte zu vereinfachen und zu beschleunigen. Aufbau und Inhalte der Datenbank werden im anschließenden Unterabschnitt 3.1.4 erklärt.

Die in der Datenbank hinterlegten Polygone müssen nach Abschluss aller Vorhersagen zusammengeführt werden. Ansonsten könnte der Fall eintreten, dass ein Gebäude, welches sich über mehrere Bildkacheln erstreckt, in Teilpolygone zerfällt (siehe Abbildung 3.5a). Da die Polygone der RasterVision Vorhersage im Bezugssystem EPSG:4326 vorliegen, werden diese zunächst nach EPSG:4839 transformiert, welches über Europa metrisch ist. Auf die transformierten Polygone wird ein Buffer von 0.1 Meter angewendet, um die durch die Kacheltrennung entstandenen Zwischenräume zu substituieren. Anschließend werden dann die überlagernden Polygone miteinander vereint. Nach der Vereinigung wird ein negativer Buffer von  $-0.1$  Metern angewendet, um die vereinten Polygone in ihre ursprüngliche Größe zurückzubringen.

Damit ist das Zusammenführen der Kachel-Vorhersagen abgeschlossen.



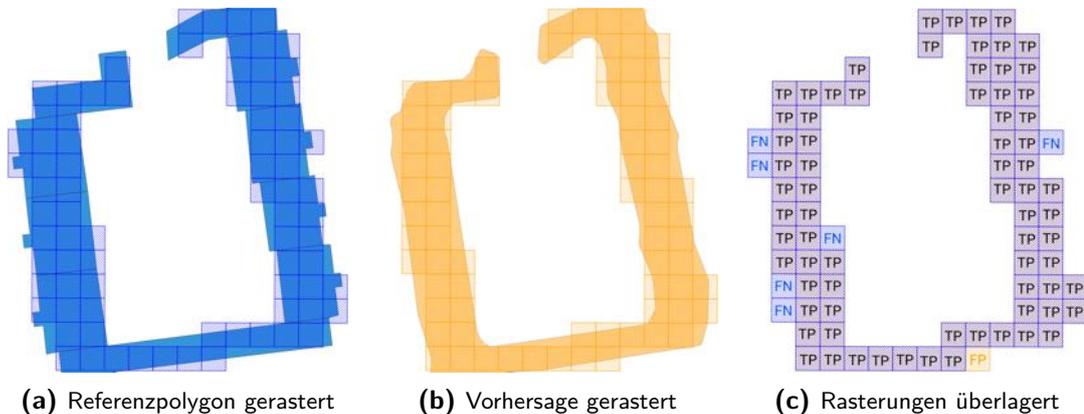
**Abbildung 3.5:** Vergleich der Polygone vor und nach Postprocessing

Da die Polygone von RasterVision auf Rastergrafiken basieren, sind die Kantenzüge der Vorhersagen entsprechend komplex. In Abbildung 3.5a lässt sich erkennen, dass die Kanten entlang der Pixel des ursprünglichen Eingabebildes verlaufen. Es entsteht ein deutlich höherer Speicherbedarf für ein Gebäude, da für jede Abzweigung eigene Koordinaten gespeichert werden müssen. Auf einer perfekten Diagonalen von  $n$  Metern wäre dies ein linearer Speicherbedarf von  $\mathcal{O}(n)$ , da je zusätzlichen Meter eine konstante Anzahl an Zwischenkoordinaten dazu kommt. Es würde jedoch ausreichen den Anfangs- und Endpunkt zu speichern, sodass ein konstanter Speicherbedarf von  $\Theta(1)$  möglich wäre. Um den Speicherbedarf zu reduzieren, wird in PostGIS der Douglas-Peucker Algorithmus auf die Polygone zur Kurvenglättung angewendet. Das deutlich weniger komplexe Polygon nach Anwendung des Algorithmus ist in Abbildung 3.5b zu sehen. Im letzten Schritt des Postprocessing werden außerdem Artefakte entfernt, die bei der Vorhersage entstanden sind. Dabei handelt es sich um kleinere Pixelbereiche, die von RasterVision als Gebäude interpretiert wurden.

Falls es für die Region einen Referenzdatensatz gibt, werden in einem weiteren Schritt die Metriken berechnet. Zur Berechnung der *accuracy* werden zunächst alle Gebäudepolygone im Referenzdatensatz gezählt ( $=: n_{all}$ ). Die Berechnung von  $n_{predicted}$  basiert auf der Idee, dass ein Gebäude genau dann erkannt wurde, wenn ein vorhergesagtes Polygon sich mit diesem schneidet. Um  $n_{predicted}$  zu berechnen, wird geprüft, ob sich ein vorhergesagter Gebäudeverband mit einem real vorhandenen Gebäude schneidet. Zum Schluss ergibt sich die bekannte Rechnung mit  $accuracy = \frac{n_{predicted}}{n_{all}}$ .

Zur Berechnung der weiteren Metriken *precision*, *recall*,  $F_\beta$  und *MCC* ist es nötig die vier Felder der binären Konfusionsmatrix zu berechnen (*true positive*, *false positive*, *true negative* und *false negative*). *positive* ist hierbei mit dem Label *building* und *negative* mit *background* gleichzusetzen. Um nun die jeweiligen Felder berechnen zu können, muss eine Aussage darüber getroffen werden, welche Flächen der jeweiligen Label richtig und welche falsch vorhergesagt wurden. Da das Berechnen von schneidenden Flächen von vielen komplexen Polygonen sehr zeitaufwendig ist, wird eine approximierte Berechnung in Kauf genommen, die im Folgenden kurz erläutert wird.

In PostGIS existiert es eine Funktionalität, mit der Vektordaten für das Speichern in ein Rasterbild vorbereitet werden können. Diese Funktion wird genutzt, um sowohl die vorhergesagten Gebäudeumrisse, als auch die Referenzdaten zu rastern. Mit Wahl der Pixelsize (in Metern) kann bestimmt werden, wie grob die Rasterung stattfinden soll. Ein Beispiel für die Rasterung mit eine Pixelgröße von 1x1 Metern ist in Abbildung 3.6 zu sehen.



**Abbildung 3.6:** Gerasterte Polygone mit 1 Meter als Pixelgröße

Zur Berechnung der benötigten Werte, werden die Quadrate des Referenzpolygons und des Vorhersagepolygons übereinander gelegt. Falls sich zwei Quadrate überlappen, handelt es sich um ein *true positive*, da sich die Vorhersage mit der Realität deckt. Quadrate, die von der Vorhersage nicht überdeckt werden, sind als *false negative* zu verstehen, da vom Modell angenommen wurde, dass sich in diesem Bereich fälschlicherweise kein Gebäude befindet. Andersherum verhält es sich, wenn das Modell fälschlicherweise ein Gebäudeteil vorhersagt, sodass es sich um ein *false positive* handelt. Alle Flächen, die nicht mit Quadranten belegt worden sind, wurden korrekt als Hintergrund klassifiziert, was als *true negative* bezeichnet wird. Das zählen und matchen der Quadrate lässt sich ohne die Geometrie umsetzen. Dafür werden die Koordinaten der oberen linken Ecke von jedem Quadrat genommen und mit einem Index in der Datenbank hinterlegt, sodass eine effiziente Vergleichsoperation über alle Werte möglich ist. Damit ist die Grundlage für die Berechnung aller weiteren Metriken abgeschlossen.

### 3.1.4 Speicherung der Ergebnisse in der Datenbank

Der Ablauf des gesamten Trainingsprozesses wird durch eine Datenbank dokumentiert. Dadurch werden die Daten verschiedenster Trainingsdurchführungen zentral verwaltet, sodass Operationen und Vergleiche unter diesen einfacher werden. In Abbildung 3.7 ist ein Auszug der Struktur der Datenbank gegeben auf dessen Konzept kurz eingegangen werden soll.

Der zentrale Entitätstyp stellt *experiment* dar. In dieser werden einige Rahmendaten zur Durchführung gespeichert z. B. wie lange ein Experiment dauert, welche Hardware eingesetzt wurde, welche Bildeinstellungen verwendet und welche AOI und OOI genutzt wurden. Die Geometrie der AOI wird in der Tabelle *region* gespeichert, da mehrere Experimente inner-

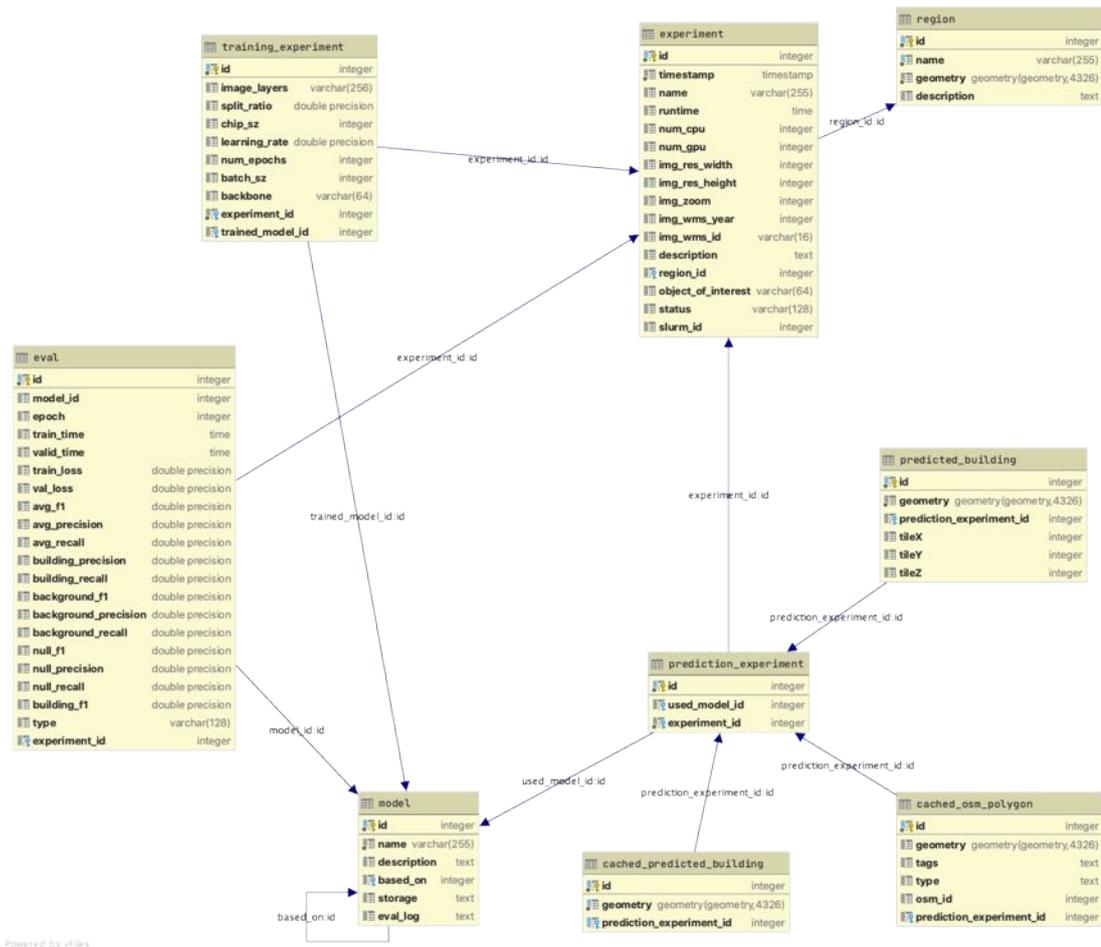
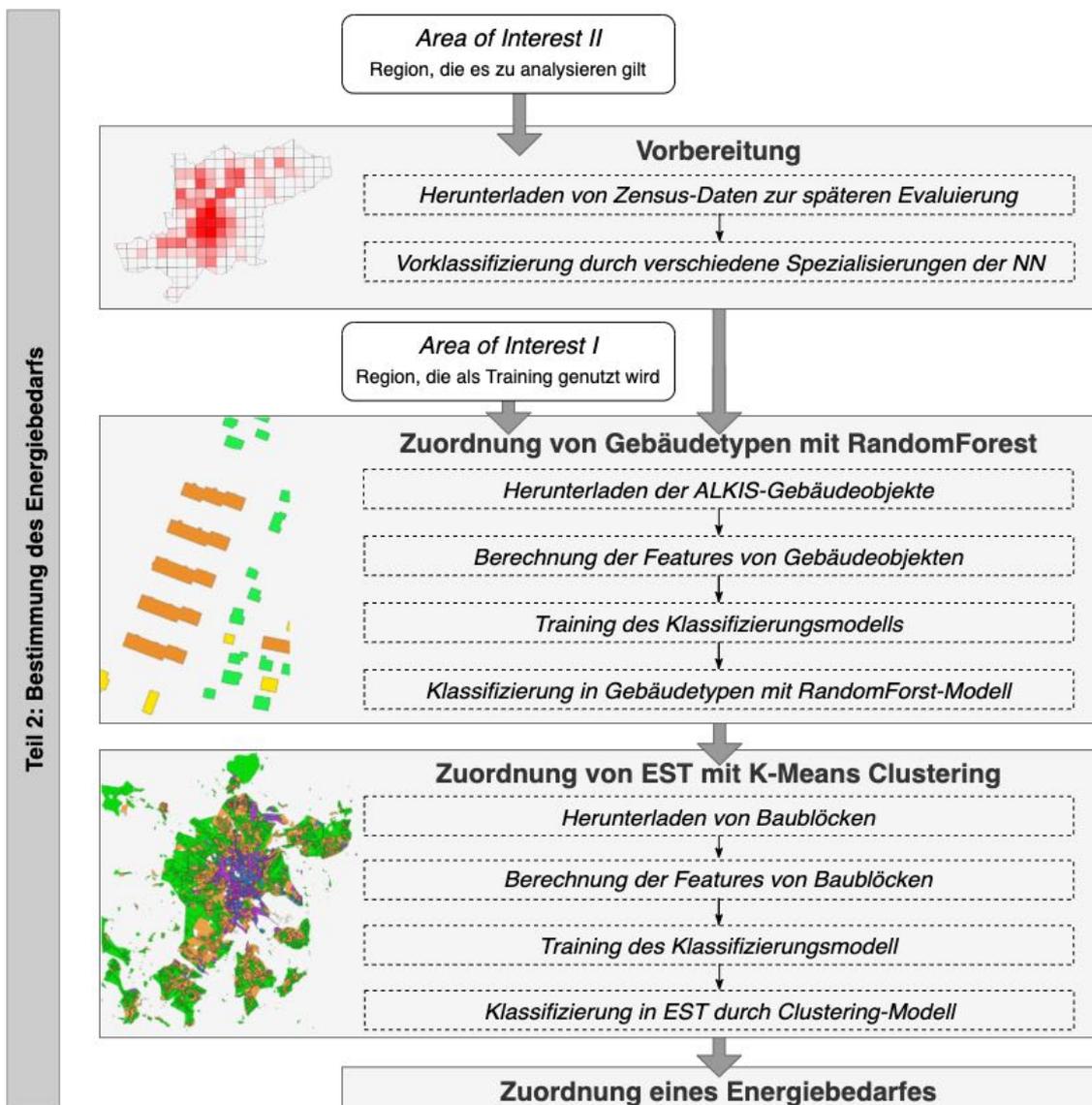


Abbildung 3.7: Auszug der wichtigsten Tabellen der Datenbankstruktur des ersten Teils

halb derselben Region stattfinden können. Auf der Experiment-Tabelle basieren die Entitäten *training\_experiment*, welche die zur Trainingsdurchführung genutzten Parameter speichert, und *prediction\_experiment*, welches das zur Vorhersage genutzte Modell speichert. So sind in *training\_experiment* die weiteren Parameter hinterlegt, die für das Training mit RasterVision verwendet wurden (Epochs, Lernrate usw.). Das durch das Training entstandene Modell wird in der Tabelle *model* referenziert. Die durch RasterVision generierten Metriken werden in *eval* abgelegt. In dieser werden die Scores je Trainingsepoch gespeichert, sodass nachvollzogen werden kann, wie sich ein NN nach mehreren Epochs entwickelt. In *prediction\_experiment* wird mit *used\_model\_id* angegeben, welches Modell für die Vorhersagen genutzt wurde. Die vorhergesagten Polygone der eingegebenen Bildkacheln werden dann als Instanz der Entität *predicted\_building* gespeichert. Nachfolgend werden die zusammengeführten und nachbearbeiteten Polygone eines Vorhersagedurchlaufes in *cached\_predicted\_building* hinterlegt. Falls Referenzdaten für die vorhergesagte Region vorhanden sind, werden diese in *cached\_osm\_polygon* gespeichert. Mit Hilfe der beiden *cached\_* Tabellen, lassen sich dann die Metriken für ein Vorhersageexperiment nach Unterabschnitt 3.1.3 berechnen.

### 3.2 Bestimmung des Energiebedarfes

Ziel des zweiten Teils der Methodik ist es den Energiebedarf der Region zu bestimmen. Dies geschieht auf der Grundlage der identifizierten Gebäudeumrisse im ersten Teil. Damit den Gebäudeumrissen ein Energiebedarf zugeordnet werden kann, muss dem umliegenden Block der energetische Stadtraumtyp zugewiesen werden. Der EST stellt Kennzahlen zur Anzahl der Bewohner, dem Verbrauch pro Bewohner und dem Verbrauch pro Grundfläche bereit [HEG20]. Um einem Block einen EST zuordnen zu können, müssen zuvor die Gebäudebauweisen der enthaltenen Gebäude bestimmt werden. Das Verfahren des zweiten Teils wird in vier Phasen unterteilt. Eine Übersicht der Phasen bietet Abbildung 3.8.



**Abbildung 3.8:** Verfahrensschema des zweiten Teils der Modellierung

Zunächst gibt es wieder eine Vorbereitungsphase, die Daten herunterlädt und aus dem vorherigen Part aufbereitet. Außerdem ist es möglich durch spezialisierte NN eine Klassifizierung der Gebäudetypen im Vorhinein vorzunehmen (Vorklassifizierung). Darauf folgend wird noch

unklassifizierten Gebäuden mit einem RandomForest-Klassifikator eine Bauweise zugeordnet. Auf Basis dieser Ergebnisse lassen sich dann mit dem K-Means Clustering Verfahren die EST bestimmen, sodass im letzten Schritt eine Zuordnung des Energiebedarfes möglich ist. Alle Phasen werden in den nachfolgenden Unterabschnitten ausführlicher aufgeschlüsselt.

### 3.2.1 Vorbereitung

Zur Vorbereitung der weiteren Schritte des zweiten Teils wird die AOI benötigt, in welcher der Energiebedarf geschätzt werden soll. So sollen zunächst Datenquellen erschlossen werden, die nachträglich zur Evaluierung der Ergebnisse dienen können. Für AOI Regionen innerhalb Deutschlands kann der Zensus als solche Datenquelle genutzt werden.

Dafür sind in einem 100 m × 100 m Gitter Zensusergebnisse zur Einwohnerzahl und zum Gebäudebestand abrufbar. Das Gitter wird in die Datenbank in eine eigene Tabelle geladen, sodass später die vorhergesagten Daten in den Zellen aggregiert werden können. Die Gitterzellen eignen sich damit für den Vergleich zwischen den erzeugten Ergebnissen dieser Arbeit mit realen Daten. [SCH19a]

Das Verfahren sieht vor, dass die Ergebnisse von zwei NN mit dem Fokus auf Gesamtgebäuden und Wohngebäuden verwendet werden. Beide berechneten Gebäudeumrisse werden als Features für die Zuordnung der Gebäudetypen mit dem RandomForest-Klassifikator benötigt. Unter dem Begriff der Vorklassifizierung ist eine vorab getroffene Zuordnung der Gebäudetypen mit einem geeigneten NN zu verstehen. Voruntersuchungen haben ergeben, dass die spezialisierten NN für EFH, DH und RH in der Lage sind die entsprechenden Bauweisen zu segmentieren. Jedoch ist die Form des erkannten Gebäudeumrisses nicht realitätsnah genug wie am linken Bild der Abbildung 3.9 zu erkennen ist. Deshalb werden die Informationen der speziellen NN auf die Gebäudepolygone einer besseren Vorhersage übertragen. Eine genauere Vorhersage generiert das NN, welches auf Gesamtgebäude trainiert wurde (mittleres Bild). Die Vorklassifizierung kombiniert die Informationen aus dem linken Bild mit dem mittleren, indem geprüft wird welche Polygone sich untereinander schneiden. Als vorläufiges Ergebnis können zu einigen Gebäuden deren Bauweisen bereits anhand der Bilderkennung zugewiesen werden (rechtes Bild).



**Abbildung 3.9:** Beispiel des Prozesses zur Vorklassifizierung [OPE21]

### 3.2.2 Zuordnung von Gebäudetypen mit RandomForest

Eine Zuordnung der Gebäudetypen zu den identifizierten Gebäuden ist notwendig, um den Energiebedarf eines Gebäudeumrisses genauer abschätzen zu können. Dies ist darauf zurückzuführen, dass eine Abhängigkeit zwischen dem Energiebedarf und Gebäudetypen festgestellt werden kann (vgl. Unterabschnitt 2.1.2). Die Vorklassifizierung erfasst aber nicht alle Gebäudeumrisse, wie sich anhand der blauen unklassifizierten Polygone im rechten Bild der Abbildung 3.9 erkennen lässt. Deshalb Bedarf es an einer weiteren Klassifikationsmethode abseits von einer Bilderkennung durch NN.

Zur Klassifizierung der Gebäudetypen der weiteren Gebäudepolygone wird der RandomForest-Algorithmus genutzt. Da es sich bei RandomForest um ein Supervised-Learning handelt, wird ein Label-Datensatz benötigt. Als Label-Datensatz eignet sich ALKIS, da zu einigen Gebäuden Bauweisen hinterlegt wurden. Als Trainingsregion für RandomForest wird die AOI aus dem ersten Teil wieder verwendet. In dieser werden die ALKIS-Gebäudeobjekte als Trainingsgrundlage heruntergeladen.

Allerdings sind die hinterlegten Gebäudebauweisen in ALKIS je nach Typ nur geringfügig vorhanden. In Abbildung 3.10 lässt sich das für die Blockrandbebauungen beobachten. Der Ausschnitt ist der Kölner Innenstadt entnommen worden. Da Blockrandbebauungen für Innenstädte typisch sind, fallen im linken Bild der Abbildung zwei Dinge auf. Es sind deutlich zu wenig Blockrandbebauungen und die Bebauungen sind lediglich partiell erfasst worden. Eine besondere Schwierigkeit stellen die partiell erfassten Gebäude dar, da die generierten Polygone des NN Gebäudeverbände sind. Dadurch kann ein RandomForest-Klassifikator, der auf reinen ALKIS-Daten trainiert wurde, keine sinnvollen Zuordnungen erzeugen.

Um den Trainingsdatensatz zu vervollständigen und mit den Ergebnissen von NN kompatibel zu machen, lässt sich dieselbe Idee wie bei der Vorklassifizierung anwenden. Die teilweise vorhandenen Informationen über die Gebäudetypen werden mit den Vorhersagen des NN für Gesamtgebäude kombiniert (mittleres Bild). Auf der einen Seite wird dadurch die gesamte Gebäudeform einer Blockrandbebauung (und natürlich anderer Bauweisen) erfasst. Auf der anderen Seite lassen sich die Features für das Training direkt auf vorhergesagten Gebäudeverbänden berechnen, wodurch die Klassifikation wiederum genauer wird.



**Abbildung 3.10:** Erzeugung der Trainingsdaten für RandomForest [OPE21][ALK21]

Neben den Labels werden noch Features benötigt, die zuvor die Bildkacheln waren. Statt Bildinformationen müssen andere Eigenschaften als Datengrundlage genutzt werden, auf derer eine Klassifikation entschieden werden kann. Die Tabelle 3.1 beinhaltet Features, die sich anhand der im ersten Teil gefunden Polygone in PostGIS berechnen lassen können [HER08]. Die meisten Merkmale basieren auf den Eigenschaften der Geometrie eines einzelnen Poly-

Name	Beschreibung	Einheit
<i>area</i>	Grundfläche des Gebäudes	m <sup>2</sup>
<i>perimeter</i>	Umfang des Grundrisses	m
<i>convex</i>	Konvexität des Gebäudes	%
<i>circular</i>	Rundheit des Gebäudes	%
<i>max_diam</i>	Größte innere Diagonale	m
<i>num_rings</i>	Anzahl der Innenringe	abs.
<i>area_rings</i>	Fläche der Innenringe	m <sup>2</sup>
<i>num_building_buffer</i>	Anzahl der Gebäude im 100 m Radius	abs.
<i>num_urban_block_buildings</i>	Anzahl der Gebäude im selben Block	abs.
<i>urban_block_buildings_area</i>	Grundfläche aller Gebäude im selben Block	m <sup>2</sup>
<i>urban_block_coverage</i>	Bebauungsdichte des zugeordneten Blocks	%
<i>rect_area</i>	Grundfläche des umschließenden Rechtecks	m
<i>area_ratio</i>	Verhältnis zwischen area und rect_area	%
<i>rect_length</i>	Länge der längeren Seite des Rechtecks	m
<i>rect_width</i>	Länge der kürzeren Seite des Rechtecks	m
<i>rect_ratio</i>	Verhältnis von rect_length und rect_width	%
<i>building_angle</i>	Neigung des Gebäudes entlang der Hauptachse	°

**Tabelle 3.1:** Mit PostGIS berechnete Features auf Grundlage der Gebäudepolygone aus Teil 1

gons. Zusätzlich ist es relevant die Umgebung einzubeziehen, da diese ebenfalls Aufschlüsse über den Gebäudetyp selbst geben kann [HER08]. Beispielsweise liegen in einem 100 m Radius eines EFH mehr einzelne Gebäudeverbände, als bei einem Blockrandbau in der Innenstadt. Neben dem 100 m Radius als Feature für die bebaute Umgebung, lässt sich der Baublock nutzen in denen die Gebäude stehen. Die offiziellen Baublöcke sind auch in ALKIS als Objektart abrufbar, jedoch ist die Datenlage nicht für alle Regionen vollständig (siehe schraffierte Flächen in Abbildung 3.11). Als Alternative lassen sich die Baublöcke über die Topologie einer Stadt bestimmen. Die Grundidee ist, dass Baublöcke durch Straßen oder Wasserwege begrenzt werden [HEG20]. Sind diese trennenden Elemente also bekannt, lassen sich die Baublöcke darüber definieren.

In einer institutsinternen Arbeit wurde dies mit den Straßenzügen aus OSM umgesetzt. Diese topologische Aufteilung nähert sich den offiziellen Baublöcken an, was sich im Vergleich mit den blauen und schraffierten Flächen in Abbildung 3.11 erkennen lässt. Um unabhängig von OSM als Datenquelle für die Straßenzüge zu sein, ist es möglich ein NN zu trainieren, welches auf die Erkennung von Straßen spezialisiert ist. Dass dies möglich ist, wird im Anhang mit Abbildung B.3 illustriert. Das dort abgebildete Ergebnis stammt von einem in der Region Köln

auf Straßen trainierten NN. Dieses erreichte in der Region Aachen eine *accuracy* von 75% und könnte eine Alternative zur OSM Datengrundlage für die Baublock-Generierung bieten.



**Abbildung 3.11:** Auszug der ALKIS (schraffiert) und generierten (blau) Baublöcke in Aachen[OPE21][ALK21]

Die in Tabelle 3.1 aufgelisteten Features werden jeweils zweimal berechnet: zum einen auf die vorhergesagten vollständigen Gebäudepolygone und zum anderen auf die Bereiche, die als Wohngebäude erkannt wurden. Damit stehen insgesamt 34 numerische Features zur Verfügung. Eine Empfehlung mit welchen Features das RandomForest-Verfahren durchgeführt werden soll ist Anhang D in beschrieben. Der trainierte Klassifikator wird auf den Polygonen der zu analysierenden AOI angewendet. Am Ende der gesamten Phase wurde allen Polygonen ein Gebäudetyp zugewiesen.

### 3.2.3 Zuordnung von Stadtraumtypen mit K-Means Clustering

Die elektrischen Energiebedarfe zu den energetischen Stadtraumtypen (siehe Tabelle A.1) werden als Berechnungsgrundlage verwendet. Damit einem Gebäude der Energiebedarf zugeordnet werden kann, muss dem Baublock, in dem das Gebäude liegt, zunächst der passende EST zugewiesen werden. Es werden wie im vorherigen Unterabschnitt die künstlich generierten Baublöcke genutzt.

Unter genauerer Analyse der EST-Kennzahlen in Tabelle A.1 fällt auf, dass bereits fünf Merkmale (siehe Tabelle 3.2) ausreichen könnten, um eine manuelle Einteilung in die EST-Klassen vorzunehmen. Dafür lässt sich händisch ein Decision Tree (siehe Abbildung B.4) erstellen, der in insgesamt sieben Klassen unterscheidet. Dabei wurden EST 7/8 aufgrund ihrer ähnlichen baulichen Eigenschaften und des ähnlichen Energieverbrauches nicht weiter getrennt. Jedoch stellte sich in ersten Versuchen heraus, dass die Einteilung anhand der gegebenen künstlichen Baublöcke und vorhergesagten Gebäudeverbände nicht zuverlässig erscheint. Dies liegt vor allem daran, dass gerade in Gebieten abseits von Stadtzentren die künstlichen Baublöcke stärker von den offiziellen abweichen und dass die EST-Kennzahlen sich auf das Nettobauland beziehen, welches nicht ohne behördliche Quellen herausgefunden werden kann.

Feature-Bezeichnung	Beschreibung	Einheit
<i>sealed_all</i>	Versiegelung des Baublocks	%
<i>area_block</i>	Gesamtfläche des Blocks	m
<i>num_all_block_buildings</i>	Anzahl der Gebäude	abs.
<i>area_avg_accom_building</i>	Durchschnittsfläche der bewohnten Gebäudeumrisse	m <sup>2</sup>
<i>num_bauweise_5</i>	Anzahl der Blockrandbebauungen	abs.

**Tabelle 3.2:** Erste Merkmale zur Einteilung in die Klassen EST 1 bis 8

Die Informationen aus Tabelle A.1 sind demnach nicht ohne weitere Anpassung auf die Baublöcke anwendbar, da die künstlichen Baublöcke und Vorhersagen zwischen verschiedenen Regionen zu stark variieren. Eine Alternative könnte es sein, ein Verfahren zu nutzen, welches weiterhin versucht die Blöcke anhand der baulichen Merkmale voneinander zu trennen. Doch ohne die Verwendung einer allgemeingültigen Regelordnung zu verwenden.

Dies lässt sich mit dem K-Means Clustering realisieren. Der Algorithmus teilt ohne einen Referenzdatensatz die gegebenen Baublöcke in eine feste Anzahl an  $k$  vorgegebenen Klassen ein. Als geeignete Klassenanzahl erscheint  $k = 4$ , sodass der K-Means-Algorithmus versucht vier voneinander trennbare Klassen zu finden. Diese vier Klassen werden am Ende den folgenden zusammengefassten EST entsprechen:

- EST 1 / 6 (EFH, MFH und dörfliche Bebauung)
- EST 2 / 3 / 4 (Reihenhäuser, Zeilenbebauungen und Hochhäuser)
- EST 5 (Blockrandbebauungen)
- EST 7 / 8 (Innenstadt und historischer Stadtkern)

Die EST Klassen 2, 3 und 3 lassen sich in ihren baulichen Eigenschaften ohne eine Höhen-Dimension nicht auseinanderhalten, da alle drei in der Ebene ähnliche länger gezogene Gebäudeverbände bilden. Auch lassen sich EST 1 und 6 nicht trennen, da dörfliche Bebauungen (EST 6) ebenfalls von vielen kleineren Gebäuden geprägt sind. Die Trennung von 7 und 8 ist aus oben genannten Gründen nicht weiter nötig.

Als Basis benötigt der Algorithmus weitere Features, die die Flächeneigenschaften der Blöcke selber, Flächeneigenschaften der Gebäude innerhalb Blöcke, Anzahl der Gebäude mit Unterscheidung in die Bauweisen und die Versiegelungsdichte einbeziehen. Dabei werden Features generiert, die jeweils angrenzende Blöcke beachten. In Tabelle A.5 sind alle Features aufgelistet, die beim K-Means Clustering in die Berechnung einbezogen werden können. Zum Abschluss dieser Phase wurde jedem Baublock einer der vier zusammengefassten EST-Klassen zugewiesen.

### 3.2.4 Zuordnung eines Energiebedarfes

Im letzten Schritt des zweiten Teils wird den Gebäudepolygonen der Energiebedarf zugeordnet. In Tabelle A.1 lassen sich die energierelevanten Kennwerte je EST-Klasse entnehmen. Jedoch

werden durch das K-Means Clustering einige EST-Klassen zusammengefasst. Der Einfachheit halber wird daher der Durchschnitt der Kennwerte der kombinierten EST-Klassen genutzt. Die Formeln zur Berechnung werden im Folgenden vorgestellt.

Sei  $EST := \{1, \dots, 8\}$  die Menge aller EST-Klassen und  $E := \{\{1,6\}, \{2,3,4\}, \{5\}, \{7,8\}\}$  die Menge der zusammengefassten EST-Klassen. Außerdem seien  $B :=$  Menge der Baublöcke und  $G :=$  Menge der segmentierten Gebäudepolygone. Des Weiteren werden sechs Abbildungen definiert:

$$\begin{aligned} A: G &\rightarrow \mathbb{Q}, g \mapsto \text{Grundfläche von } g \text{ (in m}^2\text{)} \\ \lambda: G &\rightarrow B, g \mapsto b \text{ (Gebäude } g \text{ liegt in Baublock } b\text{)} \\ \mu: B &\rightarrow E, b \mapsto e \text{ (} e \text{ ist die zusammengefasste EST-Klassen von Block } B\text{)} \\ \nu: EST &\rightarrow \mathbb{Q}, e \mapsto \text{Dachfläche je Einwohner (in m}^2\text{)} \\ \xi_1: EST &\rightarrow \mathbb{Q}, e \mapsto \text{Energiebedarf je Quadratmeter (in } \frac{\text{kWh}}{\text{m}^2\text{a}}\text{)} \\ \xi_2: EST &\rightarrow \mathbb{Q}, e \mapsto \text{Energiebedarf je Einwohner (in } \frac{\text{kWh}}{\text{m}^2\text{a}}\text{)} \end{aligned}$$

Die Bestimmung des Energiebedarfes des Gebäudes  $g$  anhand seiner Grundfläche ist wie folgt festgelegt:

$$A(g) \cdot \frac{\sum_{e \in EST'} \xi_1(e)}{|EST'|}, \text{ mit } EST' = \mu(\lambda(g)) \subset EST \quad (3.1)$$

Neben dem Energiebedarf lässt sich für das Gebäude  $g$  außerdem die Einwohnerzahl mit dem Merkmal *Dachfläche pro Einwohner* abschätzen:

$$\text{Einwohner}(g) := A(g) \div \frac{\sum_{e \in EST'} \nu(e)}{|EST'|}, \text{ mit } EST' = \mu(\lambda(g)) \subset EST \quad (3.2)$$

Somit wäre eine alternative Einschätzung des Energieverbrauches über die Anzahl der berechneten Einwohner wiederum wie folgt möglich:

$$\text{Einwohner}(g) \cdot \frac{\sum_{e \in EST'} \xi_2(e)}{|EST'|}, \text{ mit } EST' = \mu(\lambda(g)) \subset EST \quad (3.3)$$

Welche der beiden Abschätzungen sinnvoller ist, wird im Diskussion-Kapitel evaluiert. Letztendlich ist mit der Abschätzung des Energiebedarfes das in dieser Arbeit entwickelte Verfahren einmal komplett durchlaufen worden.

## 4 Darstellung und Diskussion der Ergebnisse

In diesem Kapitel werden weitere exemplarische Ergebnisse vorgestellt. Die Ergebnisse wurden, falls nicht anders angegeben, im Stadtbezirk Aachen erzeugt. Als Trainingsregion für die einzelnen Phasen wurde der Kölner Stadtbezirk verwendet. Das Flächenverhältnis zwischen beiden Regionen entspricht ungefähr 30 zu 70, sodass es für eine ML-Validierung geeignet ist. Außerdem umfassen beide AOIs großstädtische, vorstädtische und ländliche Merkmale. Eine detaillierte Beschreibung zur Erzeugung der Ergebnisse in den beiden Regionen wird in Anhang C vorgestellt. Diese werden anschließend mit Referenzdatensätzen bewertet.

Zusätzlich stellen die ersten beiden Unterkapitel Untersuchungen zu den Trainingsbedingungen (Datengrundlage, Trainingsparameter und aussagekräftigste Features) für die genutzten ML-Verfahren an. Zum Schluss wird eine Bewertung der verschiedenen Methoden für die Bestimmung des Energiebedarfs vorgenommen.

### 4.1 Exemplarische Untersuchungen der Gebäusegmentierung

Der erste Teil des Gesamtverfahrens segmentiert Gebäudepolygone. Um die Güte des Trainings eines NN für die Segmentierung zu steigern, gibt es zwei Anpassungsmöglichkeiten: die Trainingsgrundlage und die Trainingseinstellungen. Im ersten Unterabschnitt werden die Label-Datensätze als Trainingsgrundlage in ihrer Güte und in den finalen Trainingsresultaten miteinander verglichen. In der darauffolgenden Sektion werden die Parameter für das Training mit RasterVision untersucht. Am Ende wird verglichen wie NN, die jeweils in einem anderen Kontext trainiert wurden, untereinander abschneiden.

#### 4.1.1 Gegenüberstellung der verwendeten Trainingsdatensätze

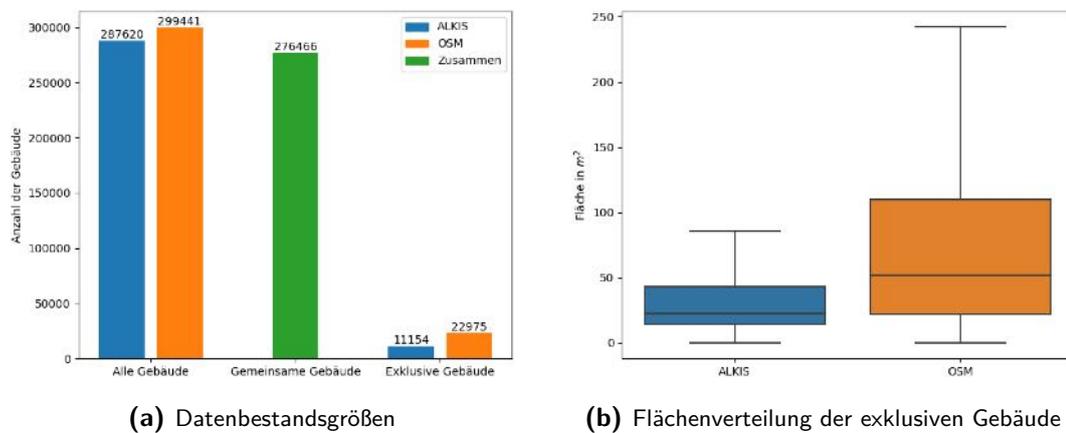
Da es für die Luftbildaufnahmen keine weiteren freien Quellen mit geeigneter Bodenauflösung, als die DOP gibt, besteht die einzige Anpassungsmöglichkeit in der Wahl des passenden Labelling-Datensatzes. In den Grundlagen wurden zum einen OpenStreetMap als globaler quelloffener Kartendienst eingeführt. Zum anderen wurde für den Raum Deutschland die behördliche Datenbank ALKIS vorgestellt. Beide stellen Gebäudepolygone bereit, die als Label für das Training eines NN genutzt werden können.

Die Datensätze werden in zwei Kategorien miteinander verglichen: Gebäudeerkennung im Allgemeinen und Erkennung von Wohngebäuden. Bei der Gebäudeerkennung im Allgemeinen (auch Gesamtgebäude genannt) soll das NN alle menschlichen Bauwerke segmentieren. Dies schließt spezielle Bauten, wie Garagen, Gartenhäuser oder Gewächshäuser mit ein. Bei der Erkennung von Wohngebäuden soll das NN erkennen, ob es sich bei einem Bauwerk um ein Gebäude zur Wohnungsnutzung handelt und ausschließlich diese segmentieren.

Um die Datensätze miteinander vergleichen zu können wird, wie in Anhang C beschrieben, als

Trainingsregion das Stadtgebiet Köln genutzt. Innerhalb der Trainingsregion wird zunächst auf die Güte der Daten von ALKIS und OSM im direkten Vergleich eingegangen. Danach wird auf der jeweiligen Label-Basis ein NN in der Region Köln trainiert. Das resultierende Modell wird auf die Region Aachen angewendet, die ebenfalls in Anhang C genutzt wurde. Zur Region Aachen stehen wieder ALKIS und OSM als Referenzdatensatz bereit, um die entsprechenden ML-Metriken für die Vorhersagen der Modelle zu erstellen.

Eine Übersicht aller erfassten Gebäude der jeweiligen Datenbestände bietet Abbildung 4.1. Die linke Grafik stellt die Anzahl der Gebäude je Datensatz dar, vergleicht sie auf gemeinsame und zeigt exklusive Gebäude auf. In der rechten Grafik werden ALKIS und OSM hinsichtlich der jeweils exklusiven Gebäudegrundflächen mit zwei Box-Plots gegenübergestellt.



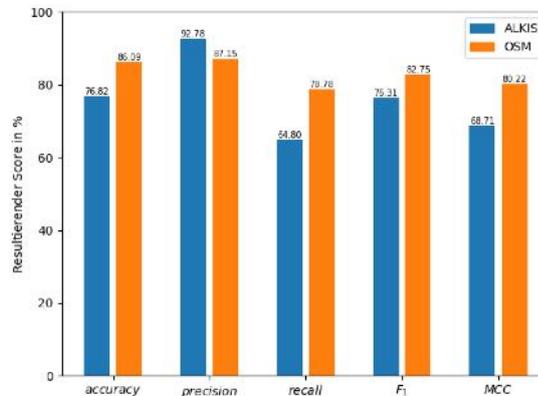
**Abbildung 4.1:** Vergleich von ALKIS und OSM bzgl. aller erfassten Gebäude in der Region Köln

OpenStreetMap stellt 299441 Gebäudepolygone bereit, während der ALKIS Datensatz in der selben Kategorie 287620 Polygone umfasst. Davon überschneiden sich 276466 Gebäude zwischen den Datensätzen. Beide Datensätze beinhalten jeweils eine größere Anzahl von Gebäuden, die im anderen nicht vorhanden sind. Um eine Abschätzung zu erhalten, welche Gebäude jeweils nicht registriert wurden, lässt sich die Verteilung der Polygonflächen, der exklusiv erfassten Gebäude, betrachten. Für die Gebäude, die ausschließlich in ALKIS vorhanden sind, beträgt der Median der Gebäudeflächen 22 m<sup>2</sup> und das obere Quartil 43 m<sup>2</sup>. Es handelt sich demnach hauptsächlich um kleinere Gebäude, wie z. B. Garagen, die nicht in OSM gelistet sind. Bei den Gebäuden, die ausschließlich in OSM enthalten sind, beträgt der Median 53 m<sup>2</sup> und das obere Quartil 110 m<sup>2</sup>.

Zusammenfassend sind in OSM etwas mehr Gebäude gelistet, die außerdem eine größere Grundfläche im Durchschnitt haben. Es ist deshalb davon auszugehen, dass der Gebäudebestand in OSM zumindest in der Region Köln genauer ist.

Es wurden nun zwei Modelle mit identischer Trainingsumgebung (selbige Parameter für RasterVision und Hardware wie in Anhang C beschrieben) trainiert, wobei eines ALKIS und das andere OSM als Trainingsgrundlage nutzte. Nach Anwendung der Modelle auf dem Stadtgebiet Aachen wurden folgende Scores berechnet, die der Abbildung 4.2 zu entnehmen sind. Dabei werden die Ergebnisse mit dem entsprechenden Referenzdatensatz verglichen, der zum Training des jeweiligen Modells genutzt wurde. Auf der X-Achse sind die Metriken *accuracy*,

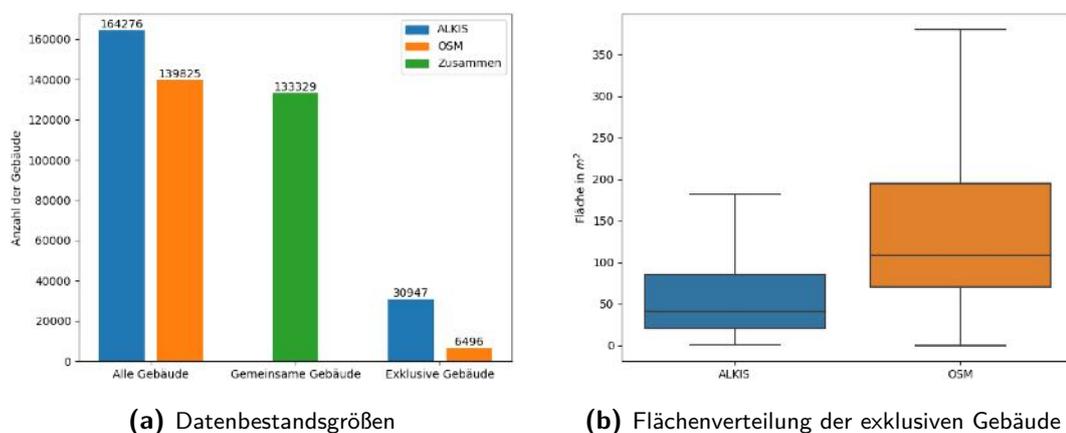
*precision*, *recall*,  $F_1$  und  $MCC$  für ALKIS (blau) und OSM (orange) untereinander gegenübergestellt worden. Auf der Y-Achse ist der resultierende Score je Metrik aufgetragen.



**Abbildung 4.2:** Berechnete Scores bei der Erkennung von allen Gebäude für ALKIS und OSM

Fast alle Ergebnisse fallen besser für das Modell aus, welches OSM als Trainingsgrundlage für alle Gebäude nutzte. Dies könnte auf den etwas größeren Gebäudebestand zurückzuführen sein, der zum Training genutzt wurde. Allein der *precision*-Score ist für ALKIS etwas höher. Dies könnte auf die umfassendere Abdeckung von kleineren Bauwerken in ALKIS zurückzuführen sein. Dafür ist die Abdeckung (*recall*) in Bezug auf allen Gebäuden deutlich schlechter als der Score für OSM. Dies resultiert in einem tieferen  $F_1$ -Score für ALKIS. Unter Einbeziehung der *background*-Klasse in der  $MCC$ -Metrik steht das ALKIS-Modell dem OSM-Modell noch weiter nach. Als Zwischenfazit scheint eine OSM Datengrundlage ein geeigneteres Modell zur Gebäudeerkennung zu trainieren.

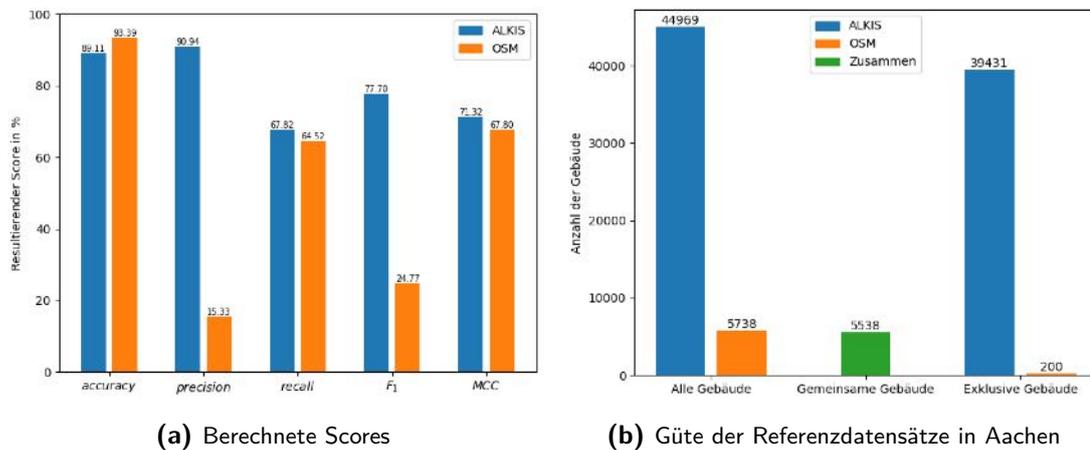
Als Nächstes soll die Kategorie der Wohngebäude betrachtet werden. Zum Vergleich werden dieselben Diagrammartentypen wie bei der Untersuchung der Gesamtgebäude genutzt. Hier scheint der ALKIS-Datensatz in der Region Köln vollständiger zu sein, wenn ausschließlich die absolute Anzahl der vorhandenen Wohngebäude in Abbildung 4.3a betrachtet wird. Bei Betrachtung der Flächenverteilung von exklusiv vorhandenen Wohngebäuden in Abbildung 4.3b fällt auf, dass die zusätzlichen Gebäude in OSM deutlich größer sind. Dies könnte damit zu-



**Abbildung 4.3:** Vergleich von ALKIS und OSM bzgl. Wohngebäude in der Region Köln

sammenhängen, dass freiwillige Beitragende in OSM einen Gebäudeverbund nicht immer in seine Bestandteile zerlegen, was bei ALKIS durch behördliche Richtlinien der Fall sein wird. Dies stellt für das Training eines Modells keinen Nachteil dar, da das Modell später ebenfalls ausschließlich einen Gebäudeverbund segmentiert und nicht die einzelnen Bestandteile trennen kann. Zusammenfassend ist die Güte des ALKIS Datensatzes in der Kategorie Wohngebäude höher als die von OSM.

Nach Training und Anwendung der Modelle ergeben sich für die Wohngebäudeerkennung die in Abbildung 4.4a abgebildeten Scores. Die höhere Qualität von ALKIS macht sich im zusammenfassenden  $MCC$ -Score leicht bemerkbar. Auffällig ist dabei, dass der  $F_1$ -Score für OSM mit 24,77% im Vergleich zu ALKIS schlecht abschneidet. Während beide einen ähnlich hohen *recall*-Wert erreichen, ist die *precision* von OSM bei 15,33%. Das OSM-Modell ordnet laut Metrik einen geringen Anteil an Gebäudepixeln präzise zu. Nach der Betrachtung von Abbildung 4.4b lässt sich diese Unstimmigkeit erklären. Es macht sich an dieser Stelle die regionale Volatilität des OSM-Datensatzes bemerkbar.



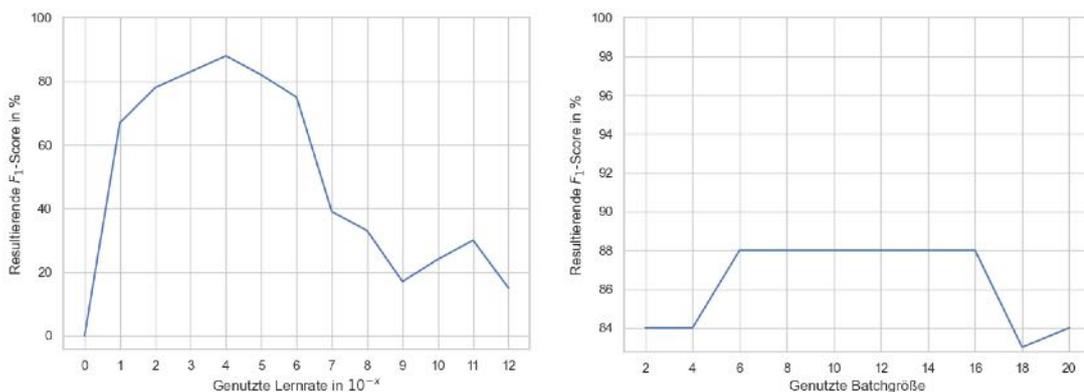
**Abbildung 4.4:** Ergebnisse der Erkennung von Wohngebäuden für ALKIS und OSM

Für die Region Aachen stellt ALKIS 44969 Gebäudepolygone zur Berechnung der Metriken bereit, während es bei OSM nur 5738 sind. Das OSM-Modell erkennt somit Gebäude in Aachen, die dem OSM-Referenzdatensatz für Aachen unbekannt sind, sodass dies fälschlicherweise als Fehler gewertet wird. Trotzdem kann davon ausgegangen werden, dass das Modell in der Region Köln ausreichend gut trainiert werden konnte. Dafür spricht zum einen der *accuracy*-Score, der angibt, dass 93,39% der Gebäude im Aachener Referenzdatensatz getroffen wurden. Zum anderen gibt dies der *recall*-Wert an, der aussagt, dass über 64% der Gebäudepixel wiedererkannt werden konnten. Abschließend lässt sich festhalten, dass sich sowohl ALKIS, als auch OSM als Trainingsgrundlage eignen, um Gebäude im Allgemeinen oder speziell Wohngebäude zu erkennen. Bei OSM muss darauf geachtet werden, dass zu Trainingszwecken eine Region mit ausreichender Datengüte gewählt wird, damit das Modell nicht zum Underfitting neigt. Außerdem lässt sich festhalten, dass sich die  $MCC$ -Metrik deutlich robuster als die  $F_1$ -Metrik verhält bei einem schlechten Referenzdatensatz.

### 4.1.2 Versuche zu den Trainingseinstellungen für RasterVision

Neben der Trainingsgrundlage lassen sich die Trainingsparameter anpassen, um das resultierende Modell zu optimieren. Dafür wurden mehrere Versuche mit unterschiedlichen Einstellungen für die Lernrate, Batchgröße, Epochs-Anzahl und des genutzten Backbones getätigt. Die Versuche haben allesamt in der Region Köln mit OSM als Labelling-Grundlage stattgefunden. Es war stets das Trainingsziel das Modell auf die Erkennung von Gebäude im Allgemeinen zu trainieren. Das Netzwerk wurde auf 80% der Bildkacheln trainiert, sodass die weiteren 20% zur Validierung genutzt werden konnten. Da der Referenzdatensatz in der entsprechenden Region vollständig ist, wurde der  $F_1$ -Score für die Gebäudeklasse genutzt, um erkennen zu können, wie gut Gebäude segmentiert werden können.

In Abbildung 4.5 sind Untersuchungen zur Lernrate und Batchgröße dargestellt. Auf der X-Achse ist jeweils die genutzte Lernrate und Batchgröße angegeben. Auf der Y-Achse ist der resultierende  $F_1$ -Score in % aufgetragen. Die Y-Achse der Lernrate beginnt bei 0% und die der Batchgröße bei 82%.



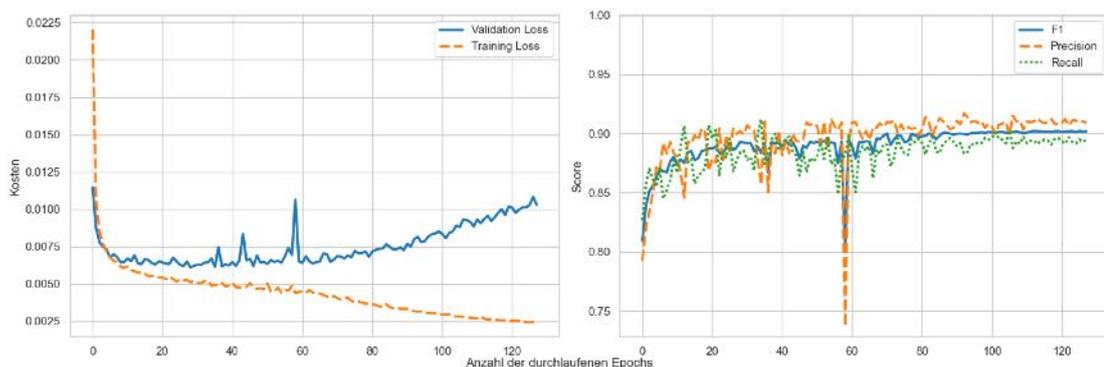
**Abbildung 4.5:** Untersuchungen zur Lernrate (links) und Batchgröße (rechts)

Die Versuche für beide Teiluntersuchungen wurden jeweils mit 5 Epochs durchgeführt. Bei den Versuchen zur Lernrate wurde zunächst eine Batchgröße von 8 festgelegt. Es lässt sich erkennen, dass eine Lernrate unter  $10^{-6}$  zu klein gewählt ist, da das Modell nach 5 vollständigen Durchläufen der Bildkacheln im Bereich des Underfitting liegt. Die zu niedrige Lernrate könnte dazu geführt haben, dass mehr Epochs nötig sind, um den vorangegangenen  $F_1$ -Score zu erreichen. Eine Lernrate von 1 ist zu hoch gewählt, da die Anpassungen der Gewichte niemals fein genug stattfinden, sodass das Modell überhaupt keine Eigenschaften zu abstrahieren lernt. Dies resultierte daher in einen  $F_1$ -Score von 0%. Letztendlich ist die Lernrate mit  $10^{-4}$  bei einem  $F_1$ -Score von 85% am Höhepunkt gewählt.

Nachfolgend wurden die Versuche zur Batchgröße mit der zuvor gefundenen Lernrate durchgeführt. Dabei konnten die Versuche nur bis zu einer Batchgröße von 20 durchgeführt werden, da die verwendete Hardware darüber in einen *Out of Memory* Fehler bei der Durchführung gelaufen ist. Bei der Batchgröße lässt sich erkennen, dass die beste Spanne im Bereich von 6 bis 16 liegt, in welcher der  $F_1$ -Score konstant bei 88% bleibt. Als Batchgröße sollte entweder

8 oder 16 gewählt werden. Die für ML entwickelte Hardware kann nämlich mit einer solchen Batchgröße, die eine Zweierpotenz ist, am effizientesten arbeiten.

Als nächstes wurde die Anzahl der Epochs bestimmt. Für das Training wurden die Lernrate  $10^{-4}$  und Batchgröße 8 gewählt. Es wurde nach jeder Epoch der aktuelle  $F_1$ -Score auf Grundlage der durchlaufenden Bilddaten berechnet. Die Berechnung des  $F_1$ -Scores findet hier auf Grundlage der Trainingsdaten und nicht Validierungsdaten statt. Der  $F_1$ -Score auf den eigenen Trainingsdaten ist bedingt aussagekräftig, da sich ein Overfitting nicht erkennen lässt. Um zu prüfen, ab welcher Anzahl von Durchläufen das Modell zum Overfitting neigt, lässt sich eine Loss-Funktion nutzen. Die Loss-Funktion bewertet wie groß die Fehlerkosten sind, die ein Modell zu einem gegebenen Datensatz macht. Dies kann z. B. in Form der totalen Quadratsumme wie beim K-Means-Clustering stattfinden. Es gilt dabei: umso kleiner die Fehlerkosten sind, umso besser sind die Vorhersagen im Allgemeinen. In Abbildung 4.6 wurde im linken Chart die Loss-Funktion für die Trainings- und Validierungsdaten nach jeder Epoch berechnet. Insgesamt wurden 128 Epochs durchlaufen. Im rechten Chart ist der *precision*-, *recall*- und  $F_1$ -Score für die Gebäudeklasse auf derselben x-Skalierung abgebildet.

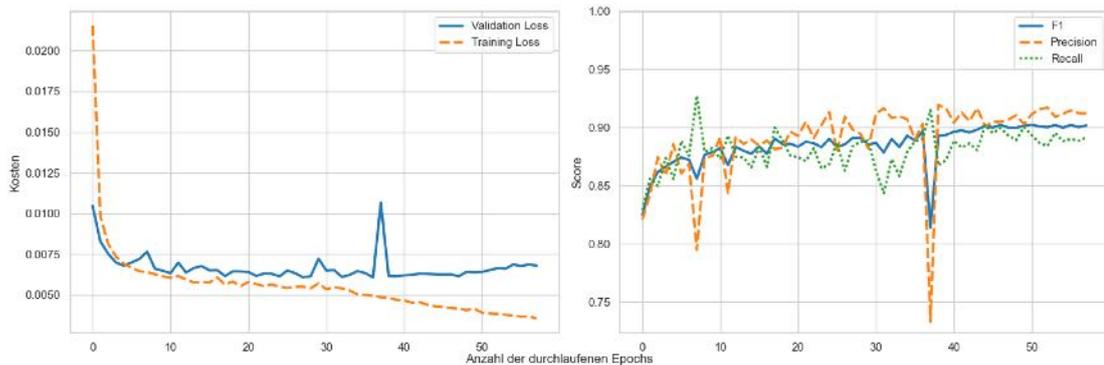


**Abbildung 4.6:** Untersuchungen zur Epochs-Anzahl mit Fehlerkosten (links) und Score (rechts)

Bei Betrachtung des rechten Charts könnte die Schlussfolgerung getroffen werden, dass die Anzahl der Epochs bei 80 gewählt werden sollte, da hier der  $F_1$ -Score bei 90% zu stagnieren beginnt. Die Funktionsergebnisse der Trainings-Loss-Kurve im linken Chart versprechen über einer Epochs-Anzahl von 80 eine stetige Verbesserung der Vorhersagen. Erst nach Einbeziehung der Validation-Loss-Kurve im linken Chart wird deutlich, dass die Entwicklung des Netzwerkes nach 60 Epochs in eine falsche Richtung verläuft. Das Modell beginnt zu overfitting, da die Fehlerkosten nach der 60. Epoch für die Validation-Loss-Funktion im Mittel stets zunehmen. Die Validation-Loss Fehlerkosten und der  $F_1$ -Score bewegen sich zwischen der 10. bis 60. Epoch in einem ähnlichen Bereich. Die stärksten Verbesserungen legt das Netzwerk im Laufe der ersten fünf Epochs hin. Die Epochs-Anzahl liegt daher im Bereich von 5 bis 20. Der maximal erreichbare  $F_1$ -Score des Modells lag bei ungefähr 89%.

Bisher wurde als Backbone ResNet50 genutzt. Im selben Setting zum Finden der Epochs-Anzahl sollte herausgefunden werden, ob es möglich ist mit ResNet101 einen höheren  $F_1$ -Score zu erreichen. Die Ergebnisse sind in Abbildung 4.7 dargestellt. Im Gegensatz zum ResNet50 hört die Skalierung der X-Achse bereits bei der 64. Epoch auf.

Es fällt auf, dass das Netzwerk bereits ab der 40. Epoch zum Overfitting neigt. Eine Anzahl der Epochs liegt hier zwischen 10 und 30. In diesem Bereich erreicht das mit ResNet101 trainierte Netzwerk ebenfalls einen maximalen Score von rund 89%. Es lässt sich daher festhalten, dass beide Backbones zum selben Ergebnis führen. Jedoch benötigte der Durchlauf je Epoch bei ResNet101 über 22 Minuten, während es beim ResNet50 nur 17 Minuten waren. Dies liegt daran, dass hier weniger Layer pro Batchdurchlauf angepasst werden mussten.



**Abbildung 4.7:** Verwendung der ResNet101 Architektur statt ResNet50.  
Links Fehlerkosten und rechts diverse Metriken

Abschließend lassen sich als optimale Trainingsparameter für RasterVision eine Lernrate von  $10^{-4}$ , eine Batchgröße von 8 und eine Epochs-Anzahl von 5 bis 20 nennen. Der Backbone kann frei gewählt werden, jedoch wurde für weitere Experimente ResNet50 aufgrund der schnelleren Trainingszeit genutzt.

#### 4.1.3 Versuche zu weiteren einflussnehmenden Eigenschaften aufs Training

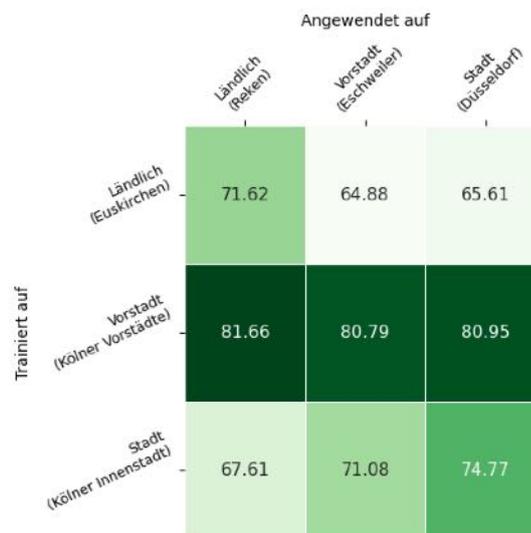
Zum Abschluss des ersten Teils soll untersucht werden, welchen Einfluss weitere Eigenschaften auf die Phase des Trainings und auf die Phase der Anwendung eines NN haben könnten. Als erstes wird untersucht, wie sich die Beschaffenheit der ausgewählten Regionen in beiden Phasen auf das NN und dessen resultierende Ergebnisse auswirkt. Die anschließenden Untersuchungen gehen auf die Eigenschaften der DOP ein, da die Qualität dieser je nach WMS variieren kann. Daher wird zunächst geprüft, welche Auswirkungen verschiedene Bodenauflösungen im Training und in der Anwendung haben. In einem weiteren Vergleich wird erprobt, ob sich ein trainiertes Modell auch dafür eignet auf anderen Bundesländern, die jeweils einen anderen WMS anbieten, angewendet zu werden.

In den nachfolgenden Versuchen war es notwendig viele verschiedene AOI zu definieren. Für das Training wurden die AOIs jeweils über einer Fläche von  $140 \text{ km}^2$  definiert. Diejenigen AOIs, auf denen das trainierte Netz angewendet wurde, wurden auf einer Fläche von  $70 \text{ km}^2$  definiert. Es wurde außerdem darauf geachtet, dass die Besiedlung zwischen den AOIs vergleichbar bleibt. Um Regionen mit einer ähnlichen Besiedlung zu finden, wurden die Daten des Zensus genutzt. Der Zensus beinhaltet Daten zur Bevölkerung in einem  $100 \text{ m} \times 100 \text{ m}$  Gitter. Diese wurden in Flächen, die größeren Postleitzahlgebieten ähneln, aggregiert und mit der Teilung durch die Flächengröße normiert, sodass die Bevölkerungsdichte gegeben ist.

Anschließend wurden die berechneten Bevölkerungsdichten auf drei Quantile gleichverteilt. Das erste Quantil umfasst somit Gebiete mit einer geringen Bevölkerungsdichte, das zweite mit einer mittleren und letzteres beinhaltet Gebiete mit einer hohen Dichte. Es ist davon auszugehen, dass die baulichen Strukturen innerhalb der jeweiligen Quantile ähnlich sind, da die genutzten Gebäudetypen mit der Einwohnerzahl korrelieren durch die EST-Klassen. Ein Beispiel für das Vorgehen findet sich im Anhang in Abbildung B.5a für die Bevölkerungsdichten in NRW.

Als Datengrundlage wurden stets alle Gebäude des OSM Datensatzes in der Trainingsregion genutzt, sodass das Netzwerk auf die Gebäudeerkennung im Allgemeinen trainiert wurde. Das Training selber wurde mit den in Unterabschnitt 4.1.2 bestimmten optimalen Trainingsparametern durchgeführt, wobei die Epochszahl bei 5 lag. Zum Vergleich der Ergebnisse wurde die *MCC*-Metrik ausgewertet. Dabei wurden die Scores jeweils in einer Matrix eingetragen dessen Y-Achse die Trainingsregion und dessen X-Achse die Anwendungsregion des NN bestimmt. Der Wert eines Feldes  $(x, y)$  in der Matrix ist dabei der *MCC*-Score einer Untersuchung mit einem Modell, welches in Region  $y$  trainiert und auf Region  $x$  angewendet wurde.

Im ersten Versuch wurde der Einfluss der Siedlungstypen auf das NN untersucht. Es wurden mit Hilfe der zuvor berechneten Bevölkerungsdichten ländliche (niedrige Dichte), vorstädtische (mittlere Dichte) und städtische (hohe Dichte) Bereiche zum Training und zur Anwendung ausgesucht. Da die Qualität der DOP sich nicht unterscheiden sollte, wurden alle AOI im Bereich NRW festgelegt, sodass derselbe WMS genutzt werden konnte. Die definierten Regionen sind in Abbildung B.5b zu finden. Die Ergebnisse dieses Versuchs sind in Abbildung 4.8 zu sehen.



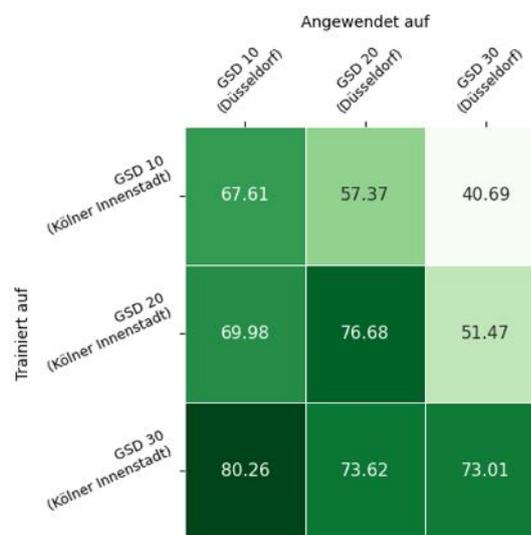
**Abbildung 4.8:** Einfluss unterschiedlicher Siedlungstypen im Training und in der Anwendung

Es fällt direkt auf, dass das NN, welches in einer vorstädtischen Region trainiert wurde, auf allen angewendeten AOIs die besten Ergebnisse erzielt. Ein Erklärungsversuch könnte sein, dass ein vorstädtisches Gebiet bauliche Elemente aus dem ländlichen und städtischen gemischt beinhaltet, sodass das Modell während des Trainings mehr Gebäudearten zu abstrahieren

lernt. Bei Betrachtung des NN, welches in einem ländlichen Gebiet trainiert wurde, lässt sich diese These stützen. Das NN schneidet nämlich nur auf der ländlichen Anwendung gut ab, während es bei den anderen beiden etwas schlechter ist. Analoge Beobachtungen gelten für das städtische NN. Dieses schneidet jedoch auf Vorstädten im Vergleich noch gut ab, da eine Vorstadt wahrscheinlich mehr städtische als ländliche Elemente vorweist.

Letztendlich lässt sich aufgrund der sehr guten Ergebnisse des vorstädtischen NN festhalten, dass eine heterogene Trainingsregion einer homogenen zu bevorzugen ist. Schließlich lässt sich das auf einer Region mit heterogener Bebauung trainierte Modell dennoch auf AOs mit homogener Bebauung anwenden, was anders herum nicht der Fall zu sein scheint.

In Abbildung 4.9 wird der Einfluss der Bodenauflösung von DOP auf die NN gezeigt. Damit sich allein die Eigenschaft der Bodenauflösung ändert und die Ergebnisse vergleichbar bleiben, wurde stets dieselbe Trainingsregion (Köln) und Anwendungsregion (Düsseldorf) verwendet. Um drei unterschiedliche Bodenaufösungen (10, 20 und  $30 \frac{\text{cm}}{\text{px}}$ ) für dieselben Gebiete zu erhalten, wurde auf das Archiv der Aufnahmen in NRW zurückgegriffen. Die Fotos zu den Auflösungen wurden 2019, 2013 und 2007 erstellt. Der Labelling-Datensatz (OSM) bietet jedoch keine Funktion um auf den Gebäudebestand in den entsprechenden Zeiträumen zugreifen zu können. Es kann davon ausgegangen werden, dass sich der Gebäudebestand in den letzten 14 Jahren nicht signifikant geändert hat, sodass der Fehler in den ausgewerteten Metriken kaum bemerkbar bleibt.

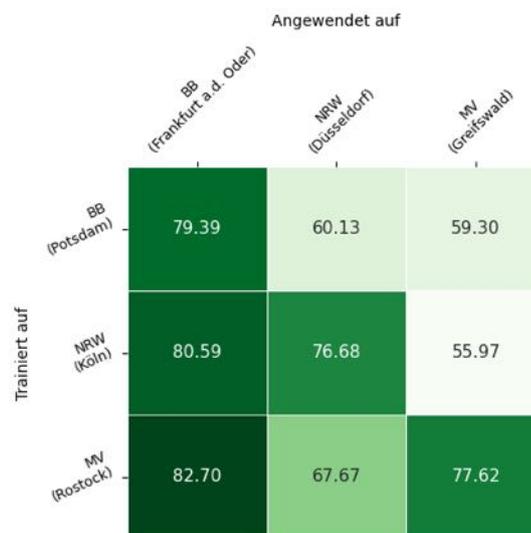


**Abbildung 4.9:** Einfluss unterschiedlicher Bodenaufösungen im Training und in der Anwendung

Ähnlich wie beim vorherigen Versuch sticht auch hier ein NN besonders hervor. Das NN, welches auf einer Bodenauflösung von  $30 \frac{\text{cm}}{\text{px}}$  trainiert wurde, schneidet in allen drei Anwendungen überdurchschnittlich gut ab. Zusätzlich fällt auf, dass das mit GSD 10 trainierte Netzwerk deutliche Schwierigkeiten beim Erkennen von Gebäuden auf Bildern mit schlechterer Auflösung hat. Dies trifft ebenfalls für das GSD 20 Netzwerk zu, welches auf GSD 30 sein schlechtestes Ergebnis erzielt. Zusammenfassend lässt sich feststellen, dass bei Möglichkeit eine etwas schlechtere Auflösung zum Training genutzt werden sollte. Das Modell wird

robuster und kann auf höheren Auflösungen einfacher Muster wiedererkennen. Anders herum scheint dies nicht der Fall zu sein.

In einer weiteren Untersuchung wurde zwischen WebMapServices mit selbiger Bodenauflösung variiert. Die folgenden Bundesländer bieten das Bildmaterial in einer Auflösung von  $20 \frac{\text{cm}}{\text{px}}$  an: Brandenburg, Mecklenburg-Vorpommern und NRW (mit Archivaufnahmen von 2013). Neben den genutzten Gebieten in NRW wurden jeweils zwei weitere Trainings- und Anwendungsregionen für die beiden zusätzlichen Bundesländer eingeführt. Dabei wurde darauf geachtet, dass die Bevölkerungsdichte der neu eingeführten Regionen denen von Köln und Düsseldorf entspricht. Die ausgewählten Regionen wurden in Abbildung B.6 dargestellt. Dies traf auf Rostock und Greifswald für Mecklenburg-Vorpommern zu. Für Brandenburg wurden Potsdam und Frankfurt an der Oder gewählt. Die Ergebnisse sind in Abbildung 4.10 aufgetragen. Entlang der Hauptdiagonalen lässt sich sehen, dass das NN auf den Bildern des zum Training



**Abbildung 4.10:** Einfluss unterschiedlicher WebMapServices im Training und in der Anwendung

genutzten WMS gut abschneidet. Dies liegt offensichtlich daran, dass die Aufnahmequalität eine ähnliche ist. Des Weiteren lässt sich anhand der Scores für NRW  $\times$  BB (80,59%) und MV  $\times$  BB (82,70%) ableiten, dass die Modelle nach ihrem Training auf Bilder von anderen WMS erfolgreich angewendet werden können. Jedoch trifft dies nicht im Allgemeinen zu, da beispielsweise das Potsdamer Modell in den Regionen der anderen beiden Bundesländer schlecht abschneidet. Dies könnte an zwei Gründen liegen: die Qualität der Aufnahmen weicht zu stark voneinander ab oder die Bebauung in der Trainingsregion Potsdam ist zu homogen gewählt. Zudem könnten architektonische Merkmale eine Rolle spielen, da Potsdam eher von historischen Gebäuden geprägt ist. Zuletzt wurden im Anhang in Abbildung B.7 die Ergebnisse eines ergänzenden Versuches angehängt. Dieser ignoriert die verschiedenen Bodenauflösungen von weiteren WMS. Die hier herausgefunden Erkenntnisse lassen sich dort wiederfinden.

## 4.2 Exemplarische Untersuchungen zu RandomForest/K-Means

In diesem Unterkapitel sollen Untersuchungen zu den verwendeten Verfahren im zweiten Teil angestellt werden. Dies betrifft den RandomForest-Algorithmus zur Zuordnung der Gebäudetypen und das K-Means-Clustering zur Einteilung in die energetischen Stadtraumtypen.

### 4.2.1 Optimierung der Zuordnung von Gebäudetypen

Die Zuordnung der Gebäudetypen lässt sich an zwei Stellen im Verfahren verbessern. Eine Möglichkeit besteht darin die Gebäudetypen mit Hilfe eines NN bereits zu erkennen, da die Bilddaten mehr Informationen bieten, als die segmentierten Gebäude. So könnte ein NN architektonische Elemente wiedererkennen, die für diese Bauweise typisch erscheinen. Die weitere Option ist eine Analyse der genutzten Features für den RandomForest-Algorithmus.

Um das Potenzial der ersten Möglichkeit zu prüfen, wurde das Netzwerk, wie im Validierungskapitel mit identischem Setting, in Köln trainiert und auf Aachen angewendet. Als Trainingsgrundlage wurde ALKIS verwendet, da sich damit die Gebäude in fünf charakteristische Bauweisen (siehe Tabelle A.2) einordnen lassen. In OSM ist zwar das Attribut `building` vergleichbar, jedoch ist der Datensatz in diesem Bezug deutlich unvollständiger als ALKIS. Die Ergebnisse für die fünf spezialisierten NN sind in Abbildung 4.11 abgebildet. Auf der X-Achse sind die Metriken *accuracy*, *precision*, *recall*,  $F_1$  und *MCC* für die 5 in ALKIS enthaltenen und überprüften Bauweisen dargestellt. Auf der Y-Achse ist der resultierende Score der jeweiligen Metrik aufgetragen.

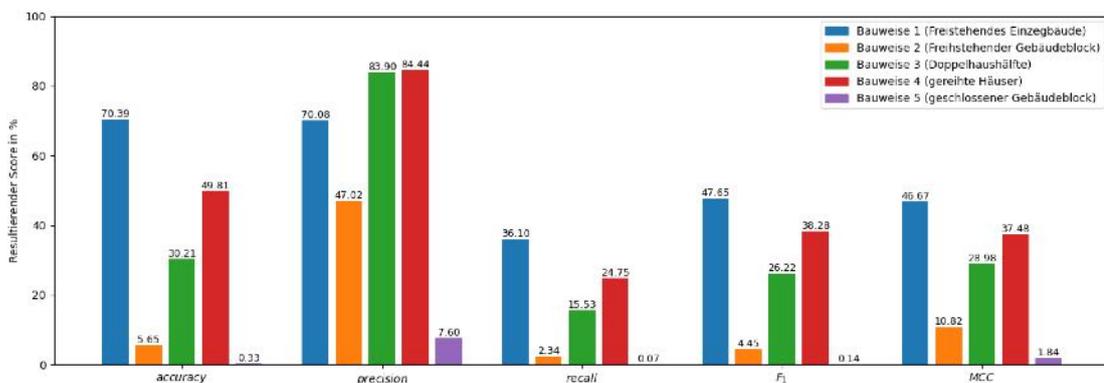


Abbildung 4.11: Ausgewertete Metriken für die Erkennung von Bauweisen durch spezialisierte NN

Zunächst einmal fällt auf, dass alle Modelle sich in einer isolierten Anwendung nicht wirklich eignen, da der *MCC*-Score überall unter 50% liegt. Für die Bauweise 5 ist der Score sogar fast bei 0, sodass das Modell hier zufällige Entscheidungen trifft, ob ein Pixel dieser Bauweise angehört, oder nicht. Die Verteilung des  $F_1$ -Scores ähnelt der des *MCC*-Scores und bestätigt damit die Aussage, dass alle Modelle im Einzelnen nicht produktiv eingesetzt werden können. Bei Betrachtung des *precision*-Scores fällt jedoch auf, dass drei Modelle über 70% liegen. Dies bedeutet, dass zwar viel zu wenig erkannt wird aufgrund des schlechten  $F_1$ -Scores, aber wenn etwas erkannt wird, dann ist nach der *precision* mit hoher Wahrscheinlichkeit ein Teil

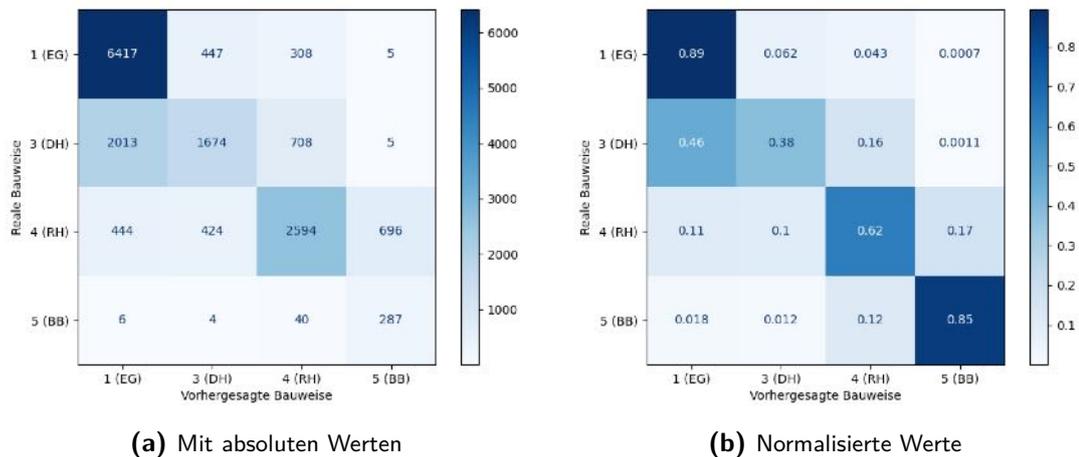
eines Gebäudes. Da der  $F_1$ -Score so niedrig ist, liegt demnach nicht an der Präzision, sondern an der schlechten Wiedererkennung der Gebäudeklasse. Der *recall*-Wert (und somit die Wiedererkennung) ist bei allen Modellen unter 40%, sodass weniger als die Hälfte der Gebäudepixel erkannt wird. Dies lässt sich für die Bauweisen 2 bis 5 in der *accuracy* sehen, die dort ebenfalls unter der Hälfte liegt, jedoch etwas höher als der *recall*. Die *accuracy* ist im Allgemeinen höher als der *recall*-Score, da der zweite noch auf die Vollständigkeit eines segmentierte Gebäudepolygons eingeht.

Zusammenfassend lässt sich folgende Schlussfolgerung ziehen: Nach den Ergebnissen der *recall*-Metrik lassen sich die segmentierten Polygone der spezialisierten NN nicht weiterverwenden. Jedoch zeigt die *precision* auf, dass einige Modelle zumindest genau bestimmen können, an welcher Position sich eine bestimmte Bauweise befindet. Im Rahmen der Vorklassifizierung kann daher auf die spezialisierten NN für die Bauweisen 1, 3 und 4 zurückgegriffen werden. Der Algorithmus der Vorklassifizierung vervollständigt dann die Polygone anhand der Ergebnisse des NN für Wohngebäude. Auf eine Vorklassifizierung der Bauweisen 2 und 5 wird verzichtet, da diese keinen ausreichenden *precision*-Wert erzielen.

Die *accuracy* sagt aus, dass durch die Vorklassifizierung nicht der gesamte Gebäudebestand einem Gebäudetyp zugeordnet werden konnte. So fehlen für die Bauweisen 1, 3 und 4 noch weitere 30%, 70% und 50% der Zuordnungen. Zudem konnten die Bauweisen 2 und 5 in der Vorklassifizierung nicht berücksichtigt werden. An dieser Stelle setzt die RandomForest-Klassifizierung ein, die in ihren Zuordnungen verbessert werden soll. In Anhang D wurden dafür Voruntersuchungen angestellt.

Insgesamt konnte das Feature-Set von 34 auf 16 Features in der endgültigen Nutzung reduziert werden. Ein weiteres Ergebnis der Voruntersuchungen war, dass es sinnvoll ist die Features auf zwei verschiedenen Grundflächenarten (Gesamtgebäude und Wohngebäude) zu berechnen. Beispielsweise sieht die reine Wohnform einer Blockrandbebauung anders aus, als das Gesamtgebäude, sodass aus beiden Polygonen unabhängig neue Informationen über die Bauweise gewonnen werden. Die bisherigen Features reichen jedoch nicht aus, um die Bauweise der Gebäudeblöcke von den gereihten Häusern zu trennen. Schließlich sind Gebäudeblöcke ebenfalls länger gezogene Häuser, sodass sie gereihten Häusern ähneln. In der Realität wären die Gebäudeblöcke jedoch höher als die gereihten Häuser, da es sich oftmals um Zeilenbebauungen mit über 8 Stockwerken handelt. Wäre eine Höhendimension im Feature-Set vorhanden, könnte eine bessere Trennung der beiden Klassen stattfinden. Da ein solches Feature nicht mit Hilfe der DOP generiert werden kann und in dem spezialisierten NN die Bauweise zu schlecht erkannt wird, muss auf die Erkennung der Gebäudeblock Bauweise verzichtet werden.

Da die Gebäudeblöcke, wie oben beschrieben, am ehesten den gereihten Häusern ähneln, werden in der finalen RandomForest-Nutzung alle Trainingsgebäude der Bauweise 2 auf Bauweise 4 gemapped. Das Ergebnis dieses Trainings ist in Form von zwei Konfusionsmatrizen in Abbildung 4.12 abgebildet. Die linke Matrix zeigt die Gebäudezuordnung in absoluten Zahlen und die rechte Matrix normalisiert die Werte, indem die Felder durch die Summe der jeweiligen Zeile geteilt werden. Auf der Hauptdiagonalen wird somit der *accuracy*-Score der jeweiligen Bauweise ausgewertet.



**Abbildung 4.12:** Konfusionsmatrizen für RandomForest-Klassifikation in 4 Bauweisen

Als Resultat konnte die korrekte Zuteilung der Bauweise 4 von 57% in den Voruntersuchungen auf 62% angehoben werden. Neben den Blockrandbebauungen (Bauweise 5) funktioniert die Erkennung der Einzelgebäude (Bauweise 1) mit fast 90% *accuracy* sehr zuverlässig. Einzig die Zuweisung der Doppelhaushälften (Bauweise 3) funktioniert mit 39% *accuracy* eher schlecht, da 46% der realen Doppelhaushälften als Einzelgebäude interpretiert werden. Dies könnte darauf zurückzuführen sein, dass kleinere Doppelhaushälften aufgrund ihrer identischen Form für die Bauweise 1 gehalten werden. Dies stellt aber weniger ein Problem dar, da in der Betrachtung der energetischen Stadtraumtypen sowieso Einzelgebäude und Doppelhaushälften sich die Klasse EST 1 und 6 teilen.

#### 4.2.2 Untersuchungen zu den Features für das K-Means-Clustering

Für das K-Means-Clustering gibt es keinen Referenzdatensatz mit dem sich die Ergebnisse validieren lassen. Stattdessen wird auf die Abbildung C.5 im Verfahrensablauf hingewiesen. Diese stellt die Zuteilung der EST-Klassen zu den Baublöcken dar. Mit Hinblick auf die Verteilung der EST-Klassen erscheint diese Verteilung zumindest visuell sinnvoll. Zwar lassen sich die Ergebnisse nicht formell auswerten, jedoch lassen sich ähnliche Voruntersuchungen wie beim RandomForest-Algorithmus durchführen. Die Voruntersuchungen zum K-Means-Clustering sind in Anhang E hinterlegt.

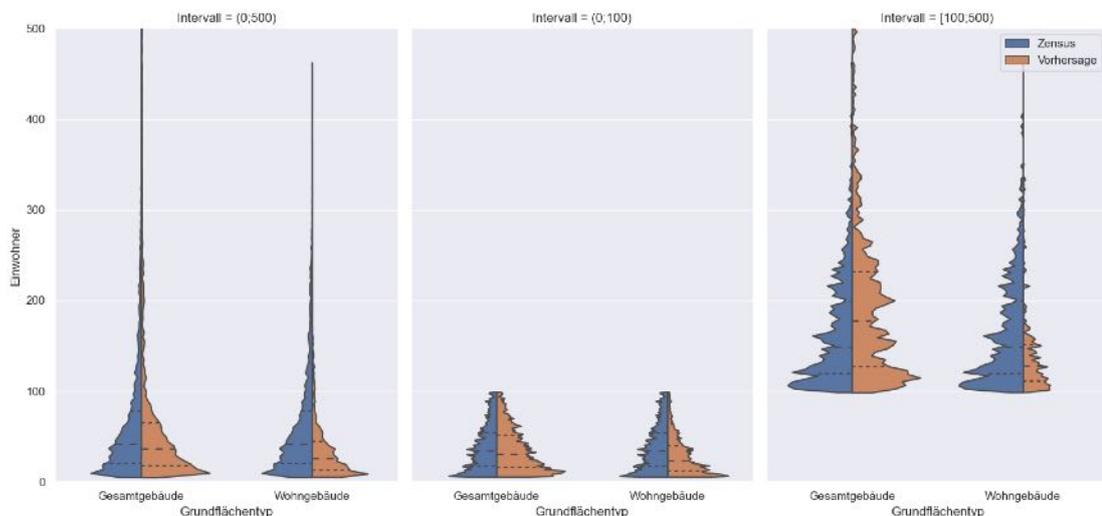
### 4.3 Exemplarische Anwendung des Verfahrens

Nachdem die Erkennung der Gebäude und Zuordnung der Bauweisen/EST optimiert wurde, soll zum Abschluss die Energiebestimmung im Detail betrachtet werden. Dabei wird zuletzt auch darüber entschieden, welche der vorgestellten Berechnungsarten in Unterkapitel Unterabschnitt 3.2.4 sich am besten eignet.

Im Verfahren unterschieden sich die Berechnungen in 2 Ebenen mit je zwei Ausprägungen. In der ersten Ebene lässt sich der Energiebedarf in einer Region entweder auf Grundlage einer

Gebäudegrundfläche oder durch die Einwohnerzahl herleiten. Die zweite Ebene unterscheidet nochmal darin, welche Gebäudegrundfläche genutzt werden soll: die gesamte Grundfläche oder nur die, die als Wohngebäude deklariert wurde. Die Berechnung der Einwohnerzahl hängt ebenfalls von der Unterscheidung in eine der zwei Grundflächentypen ab. Zur exemplarischen Betrachtung der finalen Ergebnisse des Verfahrens wurden die Ergebnisse aus dem Validierungskapitel für den Aachener Stadtbezirk genutzt.

Bevor auf die konkrete Bewertung der einzelnen Berechnungen zum Energiebedarf eingegangen wird, muss zuvor die Einwohnerzahl in einem Zwischenschritt errechnet werden. Da die Einwohnerzahl linear mit dem Energieverbrauch korreliert nach den in Unterabschnitt 3.2.4 definierten Berechnungen, ist es sinnvoll bereits diese mit einem weiteren Datensatz zu vergleichen. Für die Einwohnerzahl lässt sich als Referenzdatensatz der Zensus nutzen. Die Einwohnerzahl je Block, die nach dem K-Means-Clustering zur Verfügung gestellt werden können, müssen für einen Vergleich in das Gittersystem des Zensus umgerechnet werden. Eine visuelle Darstellung der Gitterinformationen des Zensus und einer Vorhersage bietet Abbildung B.12 im Anhang. Nachdem nun die vorhergesagten und vom Zensus modellierten Einwohnerzahlen in der Region Aachen im selben räumlichen Gitter vorhanden sind, kann die Verteilung mit Hilfe eines Violine-Plots untereinander verglichen werden. Mehrere Violine-Plots zur Untersuchung der Einwohnerzahlen sind in Abbildung 4.13 abgebildet.



**Abbildung 4.13:** Vergleich der Einwohnerverteilungen zwischen Zensus und eigenen Berechnungen

Auf der Y-Achse ist die Gitterfeldgröße in Form der Einwohnerzahl aufgetragen. Auf der X-Achse sind die genutzten Grundflächentypen (Gesamtgebäude oder Wohngebäude) angegeben, die als Grundlage für die Vorhersage genutzt wurden. Die Violinlines selbst setzen sich aus den Zensus-Daten (linke Seite/blau) und vorhergesagten Einwohnerzahlen (rechte Seite/orange) zusammen. Innerhalb der Verteilungen sind durch 3 Linien der Median und das obere/untere Quantil gegeben, sodass eine einfache Gegenüberstellung der Verteilungen möglich ist. Insgesamt wurde die Betrachtung der Gitterfelder auf drei Diagramme aufgeteilt. Im linken Diagramm wurde eine Gesamtbetrachtung der Gitterfelder dargestellt, deren Einwohnerzahl im Intervall von 0 bis 500 liegt. Im mittleren liegt eine Detailansicht auf den

dünn besiedelten Gebieten mit 0 bis 100 Einwohner vor. Das rechte Diagramm betrachtet die dichteren Gebiete mit über 100 bis 500 Einwohnern pro Feld.

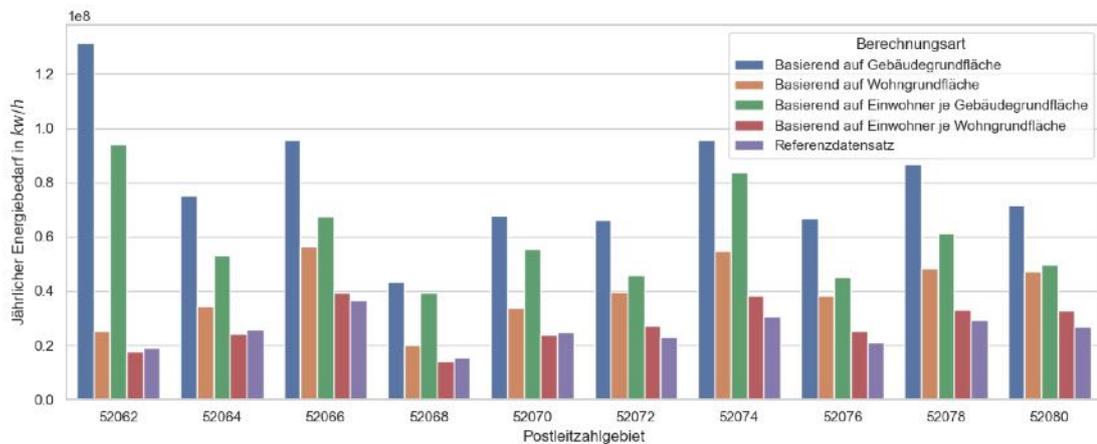
Zuerst werden die drei Violines betrachtet, die auf der Gesamtgebäude-Grundfläche basieren. Die Violine im Gesamtintervall zeigt zunächst auf, dass die Verteilung in Bezug auf Median und Quantile relativ nah an der Zensus-Datengrundlage ist. Jedoch lässt sich beobachten, dass die Grundfläche unter der vorhergesagten Verteilung viel größer ist, als die des Zensus-Datensatzes. Dies bestätigt sich auch, wenn die Gesamtbevölkerung betrachtet wird. Der Zensus gibt für den Aachener Stadtbezirk eine Bevölkerung von 234101 Einwohnern an. Die Vorhersage auf Gesamtgebäude-Grundlage liegt fast beim doppelten mit 398670 Einwohnern. Im Intervall von 0 bis 100 ergeben sich dieselben Beobachtungen bzgl. ähnlicher Verteilung und überschätzter Einwohnerzahlen. Für das letzte Intervall ergibt sich, dass die Verteilung der dicht bevölkerten Felder sich in der Vorhersage nach oben verschiebt im Vergleich zum Zensus. So liegt der Median des Zensus ungefähr bei 150 und der der Vorhersage bei 180. Dies bedeutet konkret, dass viel mehr Feldern eine hohe Bevölkerung zugeschrieben wird, als es in der Zensus-Modellierung der Fall ist. Die Erklärung dafür liegt bei den innerstädtischen Gebäudepolygonen. Diese haben eine sehr große Grundfläche, da die Bauten neben Wohnungen viele andere Einrichtungen beinhalten. Dadurch kommt es zu der vorhin aufgezeigten Überschätzung der Gesamtbevölkerung in der analysierten Region.

Als nächstes werden die weiteren drei Violines auf Basis der Wohngebäudefläche betrachtet. In der Gesamtbetrachtung ergibt sich eine kleinere Fläche unter der vorhergesagten Verteilung im Vergleich zum Zensus. Dies bestätigt sich durch die vorhergesagte Gesamtbevölkerung, die bei 182561 liegt. Damit liegt die geschätzte Einwohnerzahl um 22% unter der Echten. Im dünner besiedelten Bereich sind die Verteilungen ähnlich zueinander, sodass die Unterschätzung an anderer Stelle passiert. Die Hauptursache der Unterschätzung lässt sich am anschaulichsten in der letzten Violine finden. Es wurde nur in wenigen Feldern eine Einwohnerzahl von über 100 geschätzt. Dies könnte daraus resultieren, dass in dicht besiedelten Bereichen nicht alle Wohngebäude auf DOP als solche erkannt werden können. Gerade im innerstädtischen Bereich können Wohneinheiten beispielsweise in Gebäudekomplexen von Einkaufszentren mit integriert worden sein, sodass diese Gebäudepolygone nicht erkannt werden können. Ebenfalls ist eine Unterschätzung möglich, weil das zugehörige NN die erkannten Polygone nicht perfekt erfasst.

Zusammenfassend ist die letztere Schätzmethode auf Grundlage der Wohnfläche zu bevorzugen, da die Abweichung vom Zensus prozentual deutlich geringer ist. Durch Verbesserungen des NN für Wohngebäude ließe sich diese Abweichung eventuell noch minimieren. Dies kann mit einer Verbesserung des NN für Gebäude im Allgemeinen nicht erwartet werden kann, da die Schätzung somit noch mehr nach oben abweichen würde.

Zuallerletzt werden die vier bestimmten Energiebedarfe der Gebäude in der analysierten Region Aachen betrachtet. Um einschätzen zu können, welche Abschätzung am realistischsten ist, wird auf einen Datensatz zurückgegriffen, der den jährlichen Energiebedarf auf Postleitzahl-Ebene modelliert [CRA21]. Für den Aachener Stadtbezirk gibt es insgesamt 10 Postleitzahlgebiete, die in Abbildung B.13 angehängt wurden. Die berechneten jährliche Energiebedarfe je Block werden – ähnlich wie beim Zensusgitter – auf die entsprechenden zehn Gebiete

umgerechnet. In Abbildung 4.14 sind die vier Abschätzungen und der Referenzverbrauch je Postleitzahlgebiet zu sehen.



**Abbildung 4.14:** Berechneter Energieverbrauch in den Aachener PLZ-Gebieten im Vergleich

Die beiden Berechnungsarten, die die Gebäudegrundfläche als Grundlage nutzen, liegen stets besonders hoch über dem Referenzbedarf in den jeweiligen Postleitzahlgebieten. Im Bereich der Postleitzahl 52062 nehmen die Abschätzungen einen sechsfach höheren Verbrauch an. Da es sich bei der Postleitzahl um das Aachener Zentrum handelt, ist der Grund für die zu hohe Energieabschätzungen derselbe, der zuvor bei den Einwohnerverteilungen beschrieben wurde. Da jedes Postleitzahlgebiet dichtere Bebauungen hat, die nicht unbedingt einen Wohnnutzen haben, ist die flächenbezogene Energiebestimmung immer ein Vielfaches höher. Eine Abschätzung über die komplette Gebäudegrundfläche erscheint deshalb ungeeignet. Die verbleibenden beiden Berechnungsarten auf Basis der Wohnfläche erzeugen Ergebnisse, die näher am Referenzdatensatz sind. Dabei ist die Berechnungsweise über die Einwohnerzahl stets niedriger, als diejenige, die direkt auf der Flächenbasis arbeitet. Dies könnte daran liegen, dass die Einwohnerzahl basierend auf der Wohnfläche im Vorhinein zu gering bestimmt wurde. Die Abweichung betrug 22% weniger Einwohner als im Zensus registriert waren. Dennoch ist die Energiebedarfsbestimmung über die Einwohnerzahl näher am Referenzdatensatz dran, sodass diese Berechnungsart im Folgenden genutzt wird.

Abschließend wird der gesamte regionale Energiebedarf des Stadtbezirkes Aachen ausgerechnet. Bei Addition der Bedarfe aller zehn Gebiete ergibt sich nach dem Referenzdatensatz ein jährlicher Energiebedarf von ungefähr 253,03 GWh. Nach der geeignetsten Abschätzungsmethode ergibt sich in Summe ein Bedarf von 275,55 GWh, sodass das aggregierte Ergebnis gerade einmal 8% vom Referenzdatensatz entfernt ist.

# 5 Zusammenfassung und Ausblick

## 5.1 Zusammenfassung

Im Rahmen dieser Arbeit wurde eine Methodik zur Untersuchung der regionalen Energiebedarfe entwickelt. Diese bestimmt in zwei Schritten und nur basierend auf digitalen Orthofotos den elektrischen Energiebedarf von Gebäuden. Ein Ziel war es dabei zu gewährleisten, dass das Verfahren möglichst unabhängig von weiteren Datenquellen agieren kann, sodass der Energiebedarf auch in Regionen untersucht werden kann für die keine oder nur wenige digitale Informationen zur Verfügung stehen.

Im ersten Teil wurde unter Nutzung des Frameworks RasterVision ein Modell zur Segmentierung von Gebäuden aus Bilddaten entwickelt. Als Bildart werden DOP genutzt, da diese die Anforderung an die Auflösung erfüllten und von nationalen Behörden bereitgestellt werden. Zum Labelling werden jeweils der Kartendienst OpenStreetMap und die Plattform ALKIS genutzt. Darauf basierend werden allgemeine und spezialisierte Modelle trainiert. Die Allgemeinen können das Gesamtgebäude oder Wohngebäude segmentieren und Spezialisierte können Bauweisen erkennen. Die segmentierten Gebäude werden in Form von Polygonen zur einfacheren Weiterverarbeitung in einer PostGIS Datenbank gespeichert und ausgewertet.

Im zweiten Teil des entwickelten Verfahrens werden auf Basis der segmentierten Gebäudepolygone Energiebedarfe abgeleitet. Die Bestimmung des Energiebedarfes leitet sich aus den energetischen Stadtraumtypen ab. Dafür muss den Baublöcken, in denen sich die segmentierten Gebäude befinden, eine EST-Klasse zugeordnet werden. Um die Baublöcke zu klassifizieren, ist es nötig die Bauweisen der umschlossenen Gebäude zuvor zu bestimmen. Dies wird mit der Vorklassifizierung anhand spezialisierter Modelle und mit einem RandomForest-Klassifikator getätigt. Nach der Bestimmung der Gebäudebauweisen, lässt sich die Zuordnung der EST zu den Blöcken mit dem K-Means-Clustering durchführen und dadurch auch die Zuordnung der elektrischen Energiebedarfe zu den einzelnen Gebäuden.

Anhand exemplarischer Untersuchungen wurden unterschiedliche Parametrisierungen und Verfahrensanpassungen für die vorherigen beiden Abschnitte untersucht. Abschließend wurde das Verfahren exemplarisch auf die Region Aachen angewandt und die Ergebnisse mit den Daten eines vergleichbaren aber weniger granularen Energiesystemmodells verglichen.

Zunächst wurde die möglichen Trainingsgrundlagen des ersten Teils gegenübergestellt. OSM hat beispielsweise einen umfassenderen allgemeinen Gebäudebestand. ALKIS bietet hingegen mehr Informationen zu Wohngebäuden und ihrer Bauweise. Basierend auf der ALKIS Grundlage konnte in den Ergebnissen gezeigt werden, dass es möglich ist spezielle Gebäude zu segmentieren. Außerdem ließen sich die trainierten Modelle auf Bildern mit höherer Bodenauflösung oder von einem anderen WMS anwenden. Damit ist eine überregionale Nutzung eines trainierten Modells möglich, soweit eine heterogene Trainingsgrundlage gewählt wurde.

Es wurde ergänzend festgestellt, dass die Bestimmung der Bauweise nicht allein durch spezialisierte NN möglich ist. Diese können zwar teilweise genutzt werden, da sie in der Vorhersage pro Pixel teils sehr präzise sind, jedoch werden nicht alle Gebäude abgedeckt, sodass dieser Prozess als Vorklassifizierung bezeichnet wurde. Die vollständige Klassifizierung geschieht durch den RandomForest-Klassifikator.

Zur Vorbereitung auf das RandomForest-Training wurden verschiedenste Features auf Basis der segmentierten Polygone berechnet. In den Untersuchungen wurde erkannt, dass ausschließlich zweidimensionale Merkmale zur Verfügung stehen und daher nur 4 von 5 Bauweisen zuverlässig klassifiziert werden können. Daraus resultierte, dass die Baublöcke nur in 4 zusammengefasste EST-Klassen eingeordnet werden können statt in alle 8 Siedlungsräume. Abschließend betrug die Abweichung zu den Referenzdaten im Stadtbezirk Aachen in Bezug auf die Gesamteinwohnerzahl 22% und nur 8% bezogen auf den elektrischen Energiebedarf. Vom Endergebnis abgeleitet kann davon ausgegangen werden, dass das entwickelte Verfahren in der Lage ist realitätsbezogene Abschätzungen des jährlichen elektrischen Energiebedarfes für den Wohnsektor in einer zu analysierenden Region anzugeben.

## 5.2 Ausblick

Bisher wurden für das Verfahren ausschließlich mit WMS-Schnittstellen der deutschen Bundesländer verwendet. Um das Modell auf Regionen außerhalb Deutschlands ohne Abhängigkeit von weiteren nationalen Behörden weltweit anwenden zu können, besteht die einzige Möglichkeit in der Nutzung von hochauflösenden Satellitenbildern. Anbieter wie Planet Labs bieten Aufnahmen von jedem Ort der Welt mit einer Bodenauflösung von  $50 \frac{\text{cm}}{\text{px}}$  an [PLL21]. Die Bilder sind damit ausreichend hoch aufgelöst, sodass eine globale Alternative zu den DOP besteht. Für den Label-Datensatz zum Training reicht bereits OSM aus, da der Kartendienst eine weltweite Community hat. Jedoch muss darauf geachtet werden, eine Trainingsregion mit hinreichend vielen Gebäudedaten zu finden.

Um in Regionen mit einer weniger dichten Bebauung das Training zu verbessern, könnten beispielsweise Multipolygone verwendet werden. In den bisherigen Versuchen wurde die Trainingsregion stets über ein einziges zusammenhängendes Polygon definiert. Ein Multipolygon ist eine Ansammlung mehrerer nicht zusammenhängender Polygone. In Gebieten mit volatilen OSM-Gebäudedaten könnten so gezielt Flächen selektiert werden, die eine ausreichende Gebäudeabdeckung haben. Außerdem ist es möglich bestehende Modelle als Basis für ein weiteres Training zu nutzen, sodass mit der Zeit die Erfahrung des Modells wächst.

Zuletzt ist es möglich, weitere Object of Interests in den Fokus der Untersuchungen mit einem NN zu bringen. Anstatt Gebäude zu erkennen könnte es das Ziel sein andere spezielle Objekte, wie PV-Anlagen oder Windkraftanlagen, zu erkennen. Dies könnte dann helfen einen aktuellen Bestand der Energie-Infrastruktur innerhalb einer Region zu erfassen. Ein weiterer Anwendungsfall könnte das Zählen von Autos und Parkflächen sein, um gezielte Pläne für den Ausbau von Ladesäulen entwerfen zu können.

## Literaturverzeichnis

- [ACC17] Afonja, T.: Accuracy Paradox. 2017. URL: <https://towardsdatascience.com/accuracy-paradox-897a69e2dd9b> (besucht am 10.10.2021).
- [ALK18] Arbeitsgemeinschaft der Vermessungsverwaltungen der Länder der Bundesrepublik Deutschland (AdV): „ATKIS-Objektartenkatalog Basis-DLM“. In: *ATKIS-Katalogwerke* (2018). URL: <https://www.adv-online.de/icc/extdeu/nav/a63/binarywriterservlet?imgUid=9201016e-7efa-8461-e336-b6951fa2e0c9&uBasVariant=11111111-1111-1111-1111-111111111111>.
- [ALK21] Arbeitsgemeinschaft der Vermessungsverwaltungen der Länder der Bundesrepublik Deutschland (AdV): Amtliches Liegenschaftskatasterinformationssystem (ALKIS). 2021. URL: <https://www.adv-online.de/AdV-Produkte/Liegenschaftskataster/ALKIS/> (besucht am 08.10.2021).
- [ATK21] Arbeitsgemeinschaft der Vermessungsverwaltungen der Länder der Bundesrepublik Deutschland: Amtliches Topographisch-Kartographisches Informationssystem (ATKIS). 2021. URL: <http://www.adv-online.de/AdV-Produkte/Geotopographie/ATKIS/> (besucht am 10.05.2021).
- [AUG20] Augustyniak-Gaspers, H.: Weiterentwicklung eines Verfahrens zur Regionalisierung elektrischer Gewerbe- und Haushaltslasten auf Basis sozioökonomischer Informationen. 2020.
- [AZA21] How to Find the Latest Satellite Imagery in 2021. 2021. URL: <https://www.azavea.com/reports/how-to-find-the-latest-satellite-imagery-in-2021/> (besucht am 10.05.2021).
- [BAG19] Bundesnetzagentur: EEG in Zahlen. 2019. URL: [https://www.bundesnetzagentur.de/SharedDocs/Downloads/DE/Sachgebiete/Energie/Unternehmen\\_Institutionen/ErneuerbareEnergien/ZahlenDatenInformationen/EEGinZahlen\\_2019\\_BF.pdf?\\_\\_blob=publicationFile&v=2](https://www.bundesnetzagentur.de/SharedDocs/Downloads/DE/Sachgebiete/Energie/Unternehmen_Institutionen/ErneuerbareEnergien/ZahlenDatenInformationen/EEGinZahlen_2019_BF.pdf?__blob=publicationFile&v=2) (besucht am 02.11.2021).
- [BER21] Amtliches Liegenschaftskatasterinformationssystem (ALKIS). URL: <https://www.stadtentwicklung.berlin.de/geoinformation/liegenschaftskataster/alkis.shtml> (besucht am 08.10.2021).
- [BMU21] Bundesministerium für Umwelt, Naturschutz und nukleare Sicherheit: Fragen und Antworten zum Kohleausstieg in Deutschland. URL: <https://www.bmu.de/themen/klima-energie/klimaschutz/nationale-klimapolitik/fragen-und-antworten-zum-kohleausstieg-in-deutschland/> (besucht am 04.11.2021).
- [BRO18] Brownlee, J.: Difference Between a Batch and an Epoch in a Neural Network. 2018. URL: <https://machinelearningmastery.com/difference-between-a-batch-and-an-epoch/> (besucht am 15.10.2021).

- [CRA21] Cramer, W. u. a.: „A simulative framework for a multi-regional assessment of local energy markets—A case of large-scale electric vehicle deployment in Germany“. In: *Applied Energy* 299 (2021).
- [DOP21a] Landesamt für Geoinformation und Landesvermessung Niedersachsen: Digitale Orthophotos des ATKIS. URL: [https://www.lgln.niedersachsen.de/startseite/geodaten\\_karten/luftbildprodukte/digitale\\_orthophotos/digitale-orthophotos-des-atkis-atkis-dop-142273.html](https://www.lgln.niedersachsen.de/startseite/geodaten_karten/luftbildprodukte/digitale_orthophotos/digitale-orthophotos-des-atkis-atkis-dop-142273.html) (besucht am 31.05.2021).
- [DOP21b] Landesamt für Vermessung und Geoinformation Schleswig-Holstein: Digitale Orthophotos des ATKIS. URL: [https://www.schleswig-holstein.de/DE/Landesregierung/LVERMGEOH/Service/serviceGeobasisdaten/geodatenService\\_Geobasisdaten\\_DOP.html](https://www.schleswig-holstein.de/DE/Landesregierung/LVERMGEOH/Service/serviceGeobasisdaten/geodatenService_Geobasisdaten_DOP.html) (besucht am 31.05.2021).
- [DTS21] Scikit Learn Consortium: Decision Trees. URL: <https://scikit-learn.org/stable/modules/tree.html> (besucht am 10.10.2021).
- [EIL12] Eilenberger, S.; Braun, M.: „Herausforderungen und Lösungen für das Verteilnetz von morgen“. In: *Stuttgarter Hochspannungs-Symposium. Stuttgart. 2012.*
- [EMA18] Emanuele, R.: „Raster Vision: A New Open Source Framework for Deep Learning on Satellite and Aerial Imagery“. In: (2018). URL: <https://www.azavea.com/blog/2018/10/18/raster-vision-release/>.
- [EMF21] Forschungsstiftung Strom und Mobilkommunikation der ETH Zürich. URL: <https://www.emf.ethz.ch/de/emf-info/themen/physik/verknuepfung-von-elektrischen-und-magnetischen-feldern/wirkleistung-blindleistung-scheinleistung/> (besucht am 04.06.2021).
- [ENE19] Energiebilanzen, AGEB–Arbeitsgemeinschaft: „Anwendungsbilanzen zur Energiebilanz Deutschland“. In: *Energieverbrauch nach Energieträgern und Anwendungszwecken. Berlin: AGEB (2019).*
- [ERL21] Stadt Erlangen: Definition Nettobauland. URL: [https://www.erlangen.de/desktopdefault.aspx/tabid-1353/1773\\_read-5490/](https://www.erlangen.de/desktopdefault.aspx/tabid-1353/1773_read-5490/) (besucht am 05.10.2021).
- [ERT16] Ertel, W.; Black, N. T.: Grundkurs Künstliche Intelligenz. Springer, 2016.
- [FIS17] Fishgold, L.; Emanuele, R.: „Deep Learning for Semantic Segmentation of Aerial and Satellite Imagery“. In: (2017). URL: <https://www.azavea.com/blog/2017/05/30/deep-learning-on-aerial-imagery/>.
- [FIS21] Fishgold, L.: Raster Vision. <https://github.com/azavea/raster-vision>. 2021.
- [GEO20] Professur für Geodäsie und Geoinformatik der Universität Rostock: Definition zu Bezugssystem, Ellipsoid, WGS84, EPSG und Gauß-Krüger. 2001. URL: <http://www.geoinformatik.uni-rostock.de/> (besucht am 06.10.2021).
- [GIS21] GISGeography: What is NDVI (Normalized Difference Vegetation Index)? 2021. URL: <https://gisgeography.com/ndvi-normalized-difference-vegetation-index/> (besucht am 31.05.2021).

- [HEC13] Hecht, R.; Kunze, C.; Hahmann, S.: „Measuring completeness of building footprints in OpenStreetMap over space and time“. In: *ISPRS International Journal of Geo-Information* 2.4 (2013), S. 1066–1091.
- [HEG20] Hegger, M. u. a.: Energetische Stadtraumtypen - Strukturelle und energetische Kennwerte von Stadträumen. Stuttgart: Fraunhofer IRB Verlag, 2020.
- [HER08] Herold, H.; Schiller, G.: „Automatische Ableitung von stadtstrukturellen Grundlegenden und Integration in einem Geographischen Informationssystem, Forschungen Heft 134, Herausgeber BBR, Bonn, ISSN 1435–4659 (Schriftenreihe)“. In: (2008).
- [HOW12] Howard, B. u. a.: „Spatial distribution of urban building energy consumption by end use“. In: *Energy and Buildings* 45 (2012), S. 141–151. ISSN: 0378-7788. DOI: <https://doi.org/10.1016/j.enbuild.2011.10.061>. URL: <https://www.sciencedirect.com/science/article/pii/S037877881100524X>.
- [IPC21] The Intergovernmental Panel on Climate Change: Global Warming of 1.5 °C. URL: <https://www.ipcc.ch/sr15/>.
- [IWU20] „Deutsche Gebäudetypologie: Systematik und Datensätze“. In: *Institut Wohnen und Umwelt GmbH (IWU): Darmstadt, Germany* (2003).
- [JIA15] Jia, Y.: Object-based Land Cover Classification with Orthophoto and LIDAR Data. 2015.
- [KLE20] Klenz, A.: Entwicklung eines Modells zur Bestimmung regionalisierter Ladeprofile von Elektrofahrzeugen auf Grundlage sozioökonomischer und geografischer Parameter. 2020.
- [KME21] Scikit Learn Consortium: K-means Clustering. URL: <https://scikit-learn.org/stable/modules/clustering.html#k-means> (besucht am 09.10.2021).
- [KOS20] GeoTIFF-Allgemeine Informationen. 2020. URL: [https://kost-ceco.ch/cms/kad\\_geotiff\\_de.html](https://kost-ceco.ch/cms/kad_geotiff_de.html) (besucht am 06.06.2021).
- [LEA97] Lear, A.: „Digital orthophotography: mapping with pictures“. In: *IEEE Computer Graphics and Applications* 17.5 (1997), S. 12–14. DOI: 10.1109/38.610196.
- [LOG15] Loga, T. u. a.: „Deutsche Wohngebäudetypologie: Beispielhafte Maßnahmen zur Verbesserung der Energieeffizienz von typischen Wohngebäuden“. In: *Institut Wohnen und Umwelt GmbH (IWU): Darmstadt, Germany* 18 (2015).
- [MAT17] Matschoss, P. u. a.: „Die Integration dezentraler erneuerbarer Energien in deutsche Verteilnetze“. In: *Institute for Advanced Sustainability Studies e. V. (IASS), Potsdam* (2017).
- [NGU18] Nguyen, C. N.; Zeigermann, O.: Machine Learning kurz & gut. O'Reilly, 2018.
- [NRW21] Bezirksregierung Köln: Webdienste. 2021. URL: [https://bezreg-koeln.nrw.de/brk\\_internet/geobasis/webdienste/index.html](https://bezreg-koeln.nrw.de/brk_internet/geobasis/webdienste/index.html) (besucht am 31.05.2021).
- [OPE21] OpenStreetMap contributors: Planet dump retrieved from <https://planet.osm.org>. <https://www.openstreetmap.org>. 2021.

- [OSM21a] OpenStreetMap contributors: Zoom Levels. 2017. URL: [https://wiki.openstreetmap.org/wiki/Zoom\\_levels](https://wiki.openstreetmap.org/wiki/Zoom_levels) (besucht am 14. 10. 2021).
- [OSM21b] OpenStreetMap contributors: DE:Elemente. 2021. URL: <https://wiki.openstreetmap.org/wiki/DE:Elemente> (besucht am 05. 06. 2021).
- [OSM21c] OpenStreetMap contributors: Overpass API. 2021. URL: [https://wiki.openstreetmap.org/wiki/Overpass\\_API](https://wiki.openstreetmap.org/wiki/Overpass_API) (besucht am 06. 06. 2021).
- [OSM21d] OpenStreetMap contributors: Was ist OpenStreetMap? 2021. URL: [https://www.openstreetmap.de/faq.html#was\\_ist\\_osm](https://www.openstreetmap.de/faq.html#was_ist_osm) (besucht am 05. 06. 2021).
- [PLL21] Planet Labs: 50cm SKYSAT IMAGERY. 2021. URL: <https://www.planet.com/50cm/> (besucht am 28. 10. 2021).
- [POS21] PostGIS Dokumentation. URL: [https://postgis.net/docs/manual-2.4/postgis-de.html#spatial\\_ref\\_sys](https://postgis.net/docs/manual-2.4/postgis-de.html#spatial_ref_sys) (besucht am 06. 10. 2021).
- [POW20] Powers, D. M.: „Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation“. In: *arXiv preprint arXiv:2010.16061* (2020).
- [SAB13] Sablatnig, R.; Zambanini, S.: „Einführung in Visual Computing – Kapitel 15, Vs.13“. In: (2013).
- [SAS07] Sasaki, Y.: „The truth of the F-measure, Manchester: MIB-School of Computer Science“. In: *University of Manchester* (2007).
- [SCH16] Scheffler, J.: Verteilnetze auf dem Weg zum Flächenkraftwerk. 2016.
- [SCH19a] Schmid, T.: „Dynamische und kleinräumige Modellierung der aktuellen und zukünftigen Energienachfrage und Stromerzeugung aus Erneuerbaren Energien“. Diss. Technische Universität München, 2019.
- [SCH19b] Schwaiger, R.; Steinwendner, J.: Neuronale Netze programmieren mit Python. Rheinwerk Computing, 2019.
- [SHM19] Shmueli, B.: Matthews Correlation Coefficient is The Best Classification Metric You’ve Never Heard Of. 2019. URL: <https://towardsdatascience.com/the-best-classification-metric-youve-never-heard-of-the-matthews-correlation-coefficient-3bf50a2f3e9a> (besucht am 10. 10. 2021).
- [SPE20] Definition Georeferenzierung. 2000. URL: <https://www.spektrum.de/lexikon/geowissenschaften/georeferenzierung/5724> (besucht am 06. 10. 2021).
- [TIF21] GeoTIFF-Allgemeine Informationen. 2020. URL: <https://datatracker.ietf.org/doc/html/rfc7946> (besucht am 06. 06. 2021).
- [URB12] Hegger, M.; Dettmar, J.: Forschungsprojekt UrbanReNet, Teil I. 2009–2012.
- [WFS20] Bezirksregierung Köln: Anleitung zur Nutzung eines Web Feature Service (WFS). 2020. URL: [https://www.bezreg-koeln.nrw.de/brk\\_internet/geobasis/webdienste/anleitung\\_wfs.pdf](https://www.bezreg-koeln.nrw.de/brk_internet/geobasis/webdienste/anleitung_wfs.pdf) (besucht am 07. 10. 2021).

- [WMS20] Bezirksregierung Köln: API-Beschreibung des NRW Web Map Services. 2020. URL: [https://www.bezreg-koeln.nrw.de/brk\\_internet/geobasis/webdienste/anleitung\\_wms.pdf](https://www.bezreg-koeln.nrw.de/brk_internet/geobasis/webdienste/anleitung_wms.pdf) (besucht am 07. 10. 2021).
- [WMS21a] Bezirksregierung Köln: Geodatendienste. 2021. URL: [https://www.bezreg-koeln.nrw.de/brk\\_internet/geobasis/webdienste/geodatendienste/](https://www.bezreg-koeln.nrw.de/brk_internet/geobasis/webdienste/geodatendienste/) (besucht am 12. 06. 2021).
- [WMS21b] Bundesamt für Kartographie und Geodäsie: WMS Deutschlandmosaik aus Sentinel-2-Daten (wms\_sentinel2\_de). 2021. URL: <https://gdz.bkg.bund.de/index.php/default/wms-deutschlandmosaik-aus-sentinel-2-daten-wms-sentinel2-de.html> (besucht am 12. 06. 2021).
- [WMS21c] Open Geospatial Consortium: Web Map Service Specifications. URL: <https://www.ogc.org/standards/wms> (besucht am 06. 06. 2021).
- [YU17] Yu, H.: boundary.now. <https://github.com/haoliangyu/boundary.now>. 2017.
- [ZEN11] Statistische Ämter des Bundes und der Länder: Der registergestützte Zensus im Überblick. URL: [https://www.zensus2011.de/DE/Zensus2011/Methode/Methode\\_node.html](https://www.zensus2011.de/DE/Zensus2011/Methode/Methode_node.html) (besucht am 07. 06. 2021).
- [ZEN21] Statistische Ämter des Bundes und der Länder: Der Zensus wird in das Jahr 2022 verschoben. URL: [https://www.zensus2022.de/DE/Aktuelles/PM\\_Verschiebung\\_in\\_Kraft.html](https://www.zensus2022.de/DE/Aktuelles/PM_Verschiebung_in_Kraft.html) (besucht am 02. 11. 2021).

## **A Tabellen**

	EST 1	EST 2	EST 3	EST 4	EST 5	EST 6	EST 7	EST 8	EST 9	EST 10
Blockgröße in m <sup>2</sup>	8000	1700	16000	16000	3800	3900	4500	4000	59000	31000
Gebäudegrundfläche in m <sup>2</sup>	92	67	155	417	147	56	65	208	2050	-
Grundflächenanteil in %	14	30	27	16	54	38	74	98	35	39
Freiflächenanteil in %	82	70	73	84	46	55	14	7	65	82
Anzahl der Gebäude je Block	14	6	80	4	13	33	29	17	-	-
Anzahl der Gebäude je Hektar	14	35	17	4	37	68	114	32	-	-
Einwohneranzahl pro ha	42	90	188	277	486	170	481	485	-	-
Dachfläche in m <sup>2</sup> pro Einwohner	50	37	16	6	16	31	24	18	-	-
Wohneinheiten pro ha	21	35	104	153	270	86	260	269	-	-
Wohnfläche pro ha	1800	4500	7800	11500	20000	7500	20000	20100	-	-
Strombedarf in MWh/ha	78	116	268	395	694000	312000	751	693	-	-
Bedarf pro Einwohner in kWh	1857	1289	1426	1426	1428	1835	1561	1429	-	-
Bedarf pro Gebäude in kWh	5571	3314	15765	98759	18757	4588	6588	21656	-	-
Bedarf pro Quadratmeter in kWh	61	49	102	237	128	82	101	104	-	-

**Tabelle A.1:** Kennwerttabelle für die energetischen Stadtraumtypen 1 bis 10 basierend auf [HEG20]

<b>Verwendete Gebäudebauweise</b>	<b>ALKIS Gebäudebauweise</b>	<b>ALKIS Beschreibung</b>
Freistehendes Einzelgebäude	Freistehendes Einzelgebäude	freistehendes Wohngebäude (Einfamilienhaus)
Freistehender Gebäudeblock	Freistehender Gebäudeblock	freistehendes Wohngebäude (Mehrfamilienhaus)
Doppelhaushälfte	Doppelhaushälfte	dreiseitig freistehendes Wohnhaus mit angebauten weiteren Wohnhaus
gereichte Häuser	Reihenhaus	mehr als 2 gleichartig aneinanderggebauten Wohnhäusern
	Haus in Reihe	eines von mehr als zwei ungleichartigen, aneinanderggebauten Wohnhäusern
	Gruppenhaus	eines von mehr als 2 gleichartigen, aneinanderggebauten Wohnhäusern
geschlossener Gebäudeblock	geschlossener Gebäudeblock	eines von mehreren aneinanderggebauten Wohngebäuden im Stadtkern (Mehrfamilienhäuser)
kein Energiebedarf	Einzelgarage	einzel stehende oder angebaute einzelne Garage für ein Fahrzeug
	Doppelgarage	einzel stehende oder angebaute einzelne Garage für zwei Fahrzeuge
	Sammelgarage	einzel stehende oder angebaute einzelne Garage für mehr als zwei Fahrzeuge
	offene Halle	eine Halle bei der alle 4 Seiten offen sind
keine Zuordnung	Sonstige	deckt sich nicht mit den hier aufgeführten Arten

**Tabelle A.2:** Gebäudebauweisen mit Beschreibung nach ALKIS Katalog und die zusammengefasste genutzte Variante im Kontext der Arbeit [ALK18]

Bundesland	Auflösung <small>in cm/px</small>	Aktualität	Lizenz	Kosten
Baden-Württemberg	10	2020	-	gegen Entgeld
Bayern	80	2018	CC3	frei
Berlin	20	2018	dl-de/by-2-0	frei
Brandenburg	20	2018	dl-de/by-2-0	frei
Bremen	8	2019	-	gegen Entgeld
Hamburg	20	2020	dl-de/by-2-0	frei
Hessen	20	2020	-	gegen Entgeld
Mecklenburg-Vorpommern	20	2020	dl-de/by-2-0	frei
Niedersachsen	40	2021	-	gegen Entgeld
Nordrhein-Westfalen	10	2021	dl-de/zero-2-0	frei
Rheinland-Pfalz	40	2020	dl-de/by-2-0	frei
Saarland	20	2021	speziell	auf Anfrage
Sachsen	20	2019	dl-de/by-2-0	frei
Sachsen-Anhalt	20	2019	dl-de/by-2-0	frei
Schleswig-Holstein	20	2019	-	gegen Entgeld
Thüringen	20	2020	dl-de/by-2-0	frei

**Tabelle A.3:** Vergleich der Auflösung, Aktualität und Kosten des DOP-Angebotes der Bundesländer

OOI	Ausgeführte Query
buildings	way[building];
accommodation	way['building'~'residential house terrace detached apartments bungalow detached semidetached_house']; way['building:use'~'residential apartments'];
commercial	way['building'~'commercial kiosk office supermarket retail']; way['shop'='supermarket']; way['landuse'='retail'];
industrial	way['building'~'industrial warehouse']; way['landuse'='industrial']; way['man_made'='works'];
religious	way['building'~'cathedral chapel church monastery mosque presbytery religious synagogue'];
streets	way['highway'~'motorway trunk primary motorway_link trunk_link primary_link unclassified tertiary secondary track path residential service secondary_link tertiary_link'];

**Tabelle A.4:** Hauptteil der an die Overpass API gestellten Query-Anfragen

Feature-Bezeichnung	Beschreibung	Einheit
Flächeneigenschaften der Blöcke		
<i>area_block</i>	Grundfläche des Blocks	m <sup>2</sup>
<i>area_block_nn</i>	Grundfläche des Blocks inkl. Nachbarblöcke	m <sup>2</sup>
Flächeneigenschaften der Gebäude innerhalb Blöcke		
<i>area_accom_buildings</i>	Fläche aller Wohngebäude	m <sup>2</sup>
<i>area_all_buildings</i>	Fläche aller Gebäude	m <sup>2</sup>
<i>area_accom_buildings_nn</i>	Fläche aller Wohngebäude inkl. Nachbarblöcke	m <sup>2</sup>
<i>area_all_buildings_nn</i>	Fläche aller Gebäude inkl. Nachbarblöcke	m <sup>2</sup>
<i>area_avg_accom_buildings</i>	Durchschnittliche bewohnte Grundfläche im Block	m <sup>2</sup>
<i>area_avg_all_buildings</i>	Durchschnittliche Grundfläche im Block	m <sup>2</sup>
<i>area_median_accom_buildings</i>	Median der bewohnten Grundflächen im Block	m <sup>2</sup>
<i>area_median_all_buildings</i>	Median aller Grundflächen im Block	m <sup>2</sup>
Anzahl der Gebäude mit Unterscheidung in die Bauweisen		
<i>num_accom_block_buildings</i>	Anzahl aller Wohngebäude	abs.
<i>num_all_block_buildings</i>	Anzahl aller Gebäude	abs.
<i>num_accom_block_buildings_nn</i>	Anzahl aller Wohngebäude inkl. Nachbarblöcke	abs.
<i>num_all_block_buildings_nn</i>	Anzahl aller Gebäude inkl. Nachbarblöcke	abs.
<i>num_bauweise_1</i>	Anzahl aller Gebäude mit ALKIS Bauweise 1	abs.
<i>num_bauweise_3</i>	Anzahl aller Gebäude mit ALKIS Bauweise 3	abs.
<i>num_bauweise_4</i>	Anzahl aller Gebäude mit ALKIS Bauweise 4	abs.
<i>num_bauweise_5</i>	Anzahl aller Gebäude mit ALKIS Bauweise 5	abs.
<i>num_bauweise_1_nn</i>	<i>num_bauweise_1</i> inkl. Nachbarblöcke	abs.
<i>num_bauweise_3_nn</i>	<i>num_bauweise_3</i> inkl. Nachbarblöcke	abs.
<i>num_bauweise_4_nn</i>	<i>num_bauweise_4</i> inkl. Nachbarblöcke	abs.
<i>num_bauweise_5_nn</i>	<i>num_bauweise_5</i> inkl. Nachbarblöcke	abs.
Versiegelungsdichte		
<i>sealed_accom</i>	Versiegelung des Blocks durch Wohnbebauungen	%
<i>sealed_all</i>	Versiegelung des Blocks durch alle Gebäude	%
<i>sealed_accom_nn</i>	<i>sealed_accom</i> inkl. Nachbarblöcke	%
<i>sealed_all_nn</i>	<i>sealed_all</i> inkl. Nachbarblöcke	%

Tabelle A.5: Geeignete Features für das K-Means Clustering

## **B Bilder**

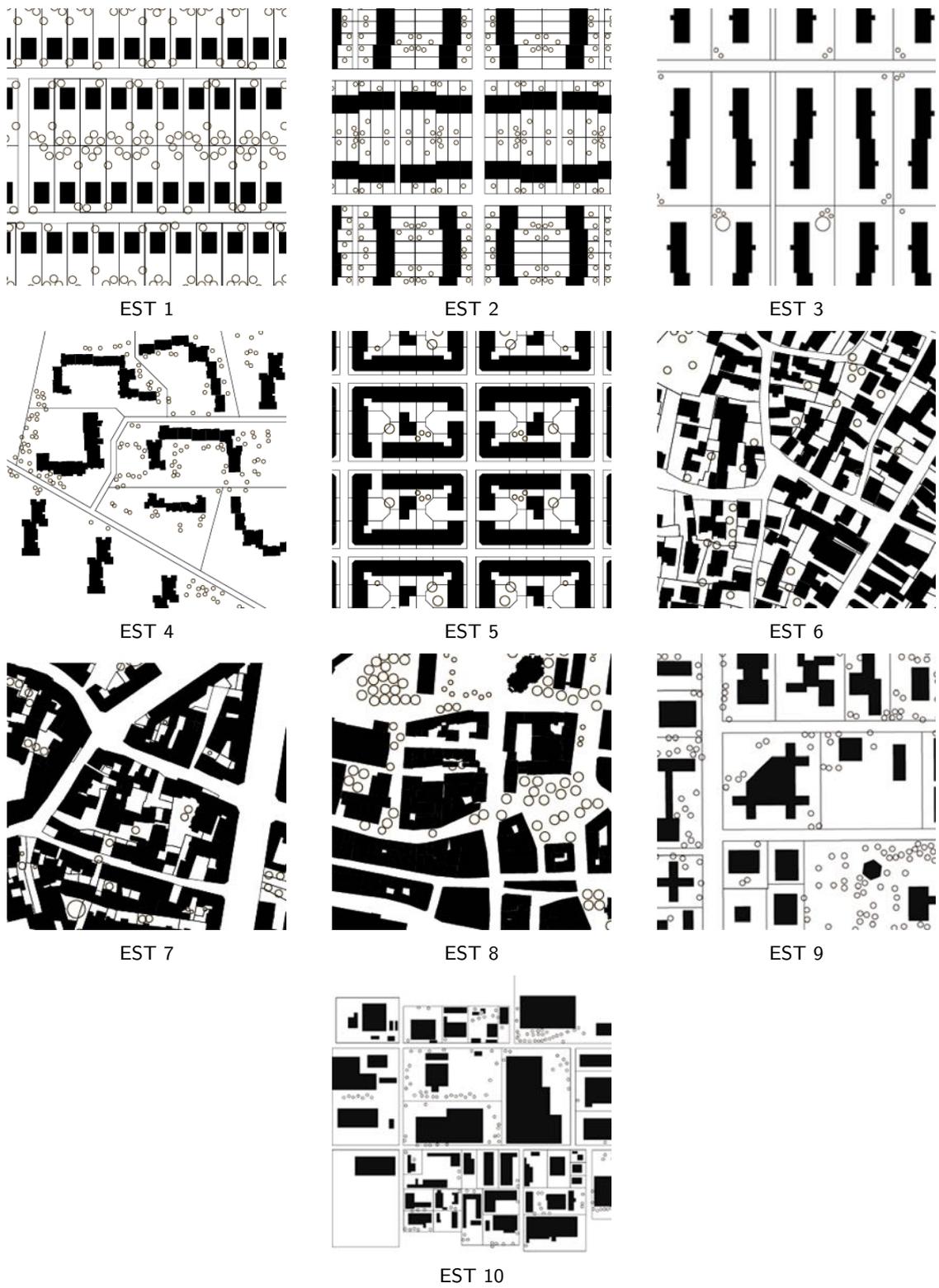


Abbildung B.1: Die Siedlungsräume aus Strukturperspektive von oben betrachtet [HEG20]



**Abbildung B.2:** Gegenüberstellung von TrueDOP mit DOP, historischen Aufnahmen und verschiedenen Bodenauflösungen des NRW Web Map Services. Bei den normalen DOP Bildern erscheinen einige Häuser gekippt, sodass die seitliche Fassade sichtbar ist. [WMS21a]



**Abbildung B.3:** Durch ein neuronales Netz erkannte Straßenzüge in der Region Aachen

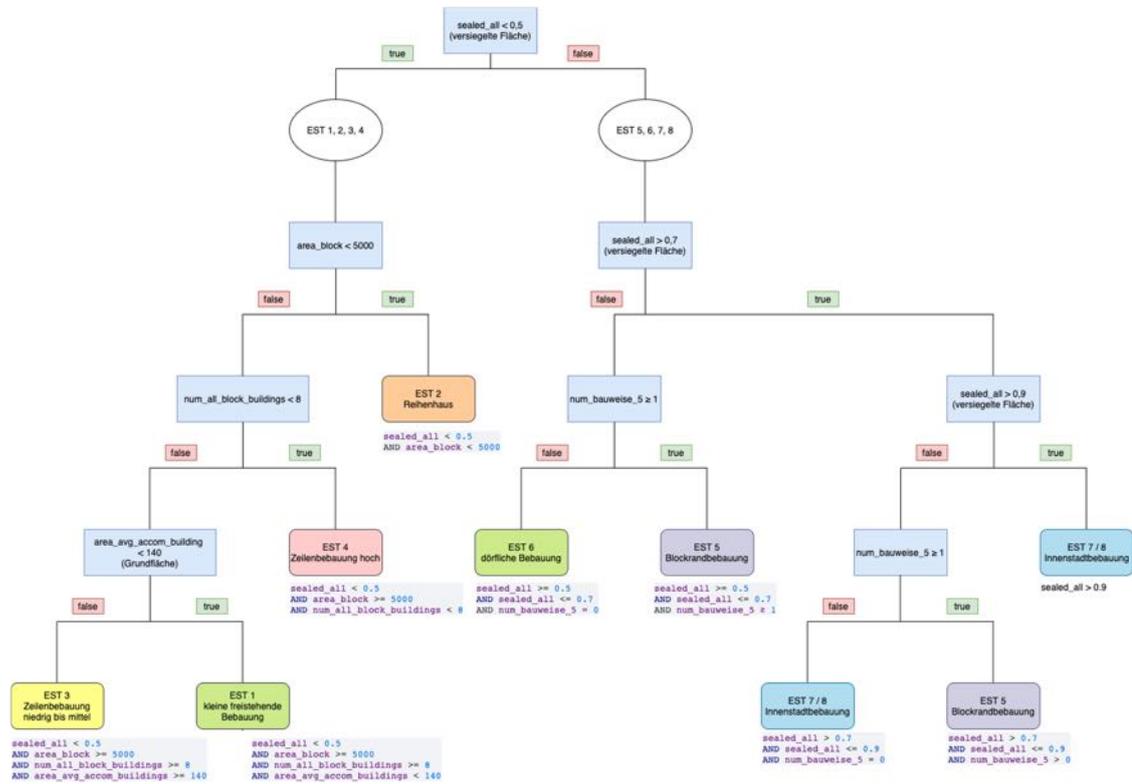
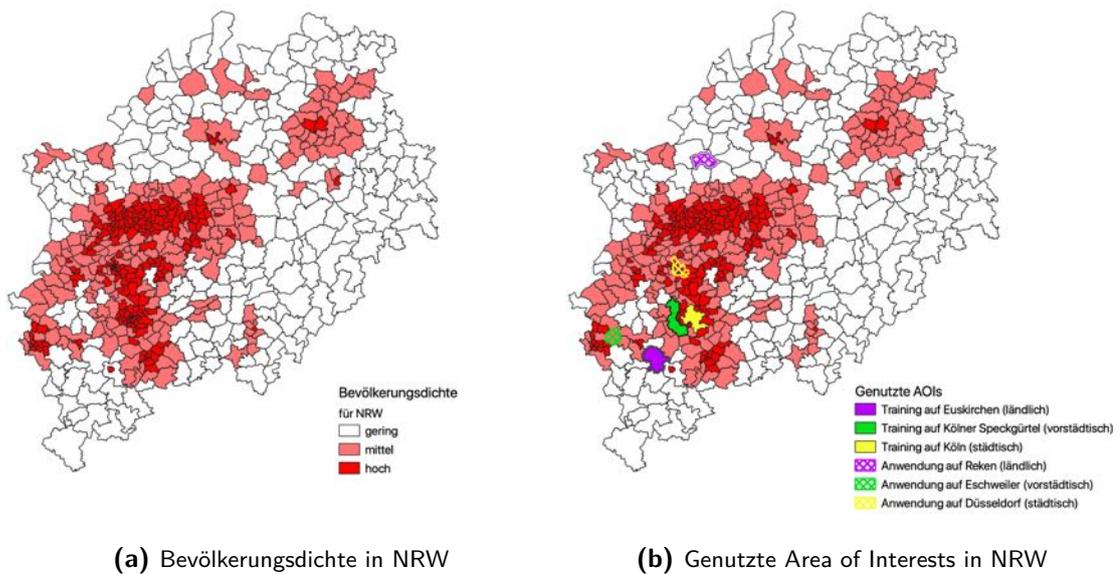


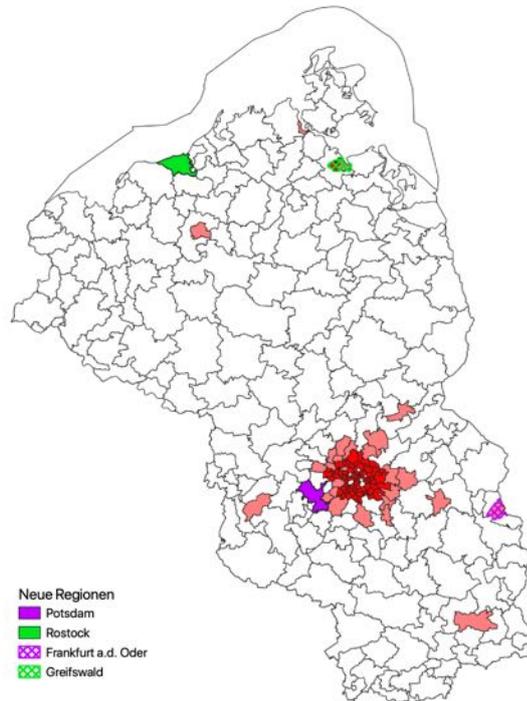
Abbildung B.4: Manuell erstellter Decision Tree auf Grundlage der EST-Kennzahlen



(a) Bevölkerungsdichte in NRW

(b) Genutzte Area of Interests in NRW

Abbildung B.5: Darstellung der Vorgehensweise beim Auswählen der AOIs für die exemplarischen Untersuchungen



**Abbildung B.6:** Weitere eingeführte Regionen für WMS Untersuchungen bei selber Bodenauflösung

Trainiert auf	Angewendet auf			
	NRW   GSD 10 (Düsseldorf)	BB   GSD 20 (Frankfurt a.d. Oder)	RP   GSD 40 (Trier)	BY   GSD 80 (Ingolstadt)
NRW   GSD 10 (Köln)	67.61	72.56	34.17	38.34
BB   GSD 20 (Potsdam)	56.17	79.39	63.48	65.62
RP   GSD 40 (Koblenz)	62.18	76.04	70.32	72.17
BY   GSD 80 (Nürnberg)	68.21	71.92	55.23	69.54

**Abbildung B.7:** Vergleichsmatrix von Web Map Services mit verschiedener Bodenauflösungen

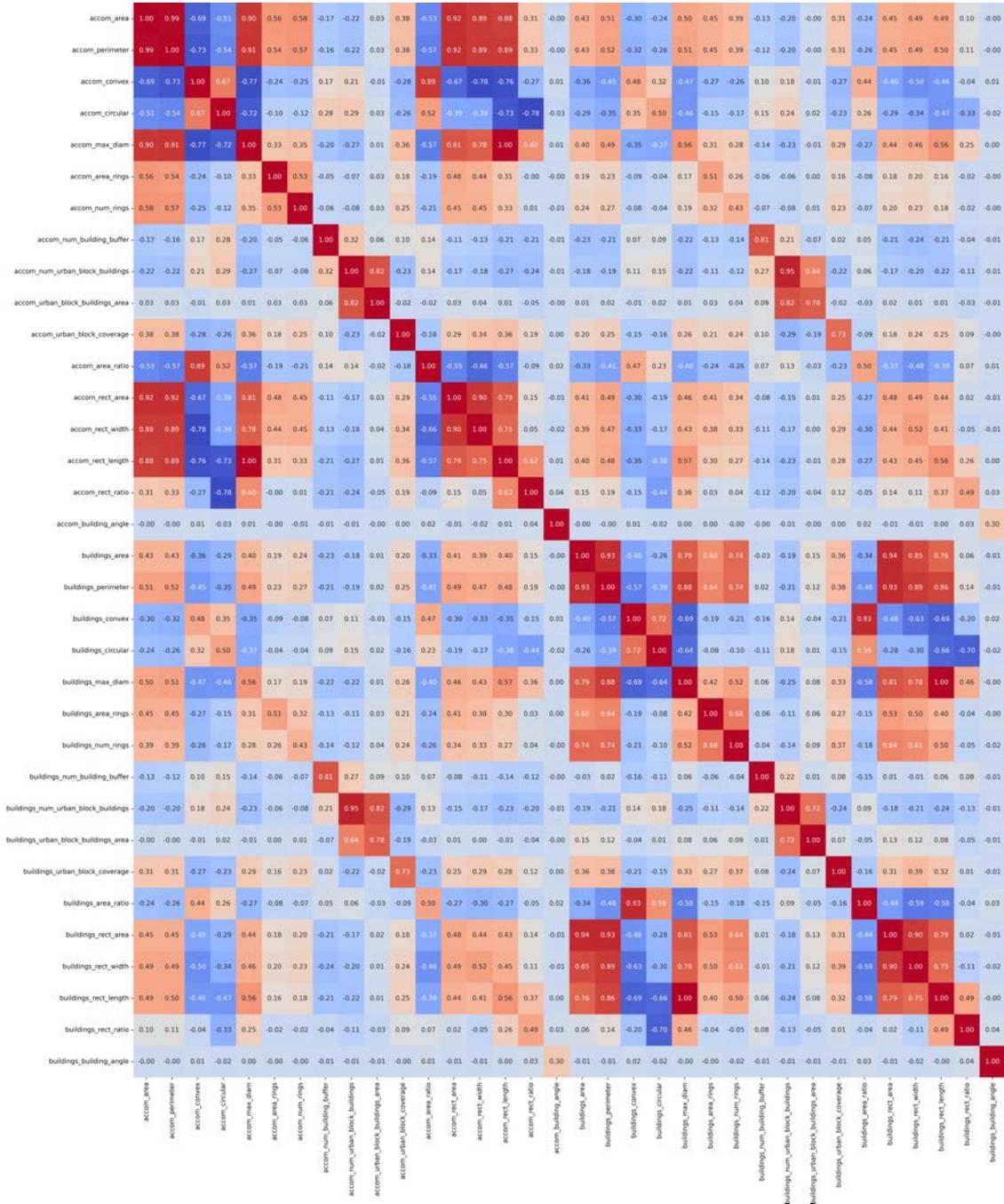


Abbildung B.8: Vollständige Korrelationsmatrix aller 34 Features für RandomForest

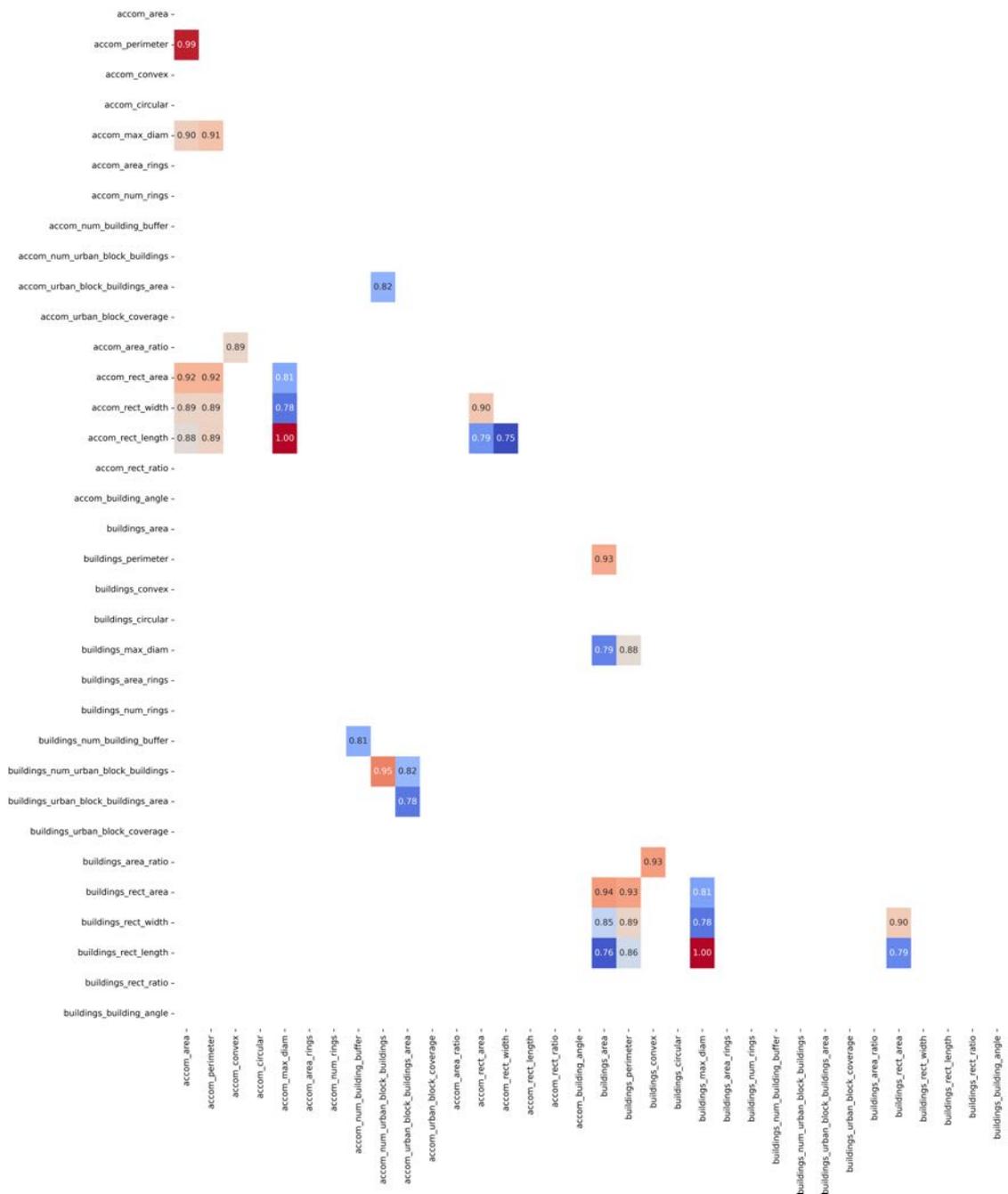


Abbildung B.9: Korrelationsmatrix der stark korrelierenden Features für RandomForest

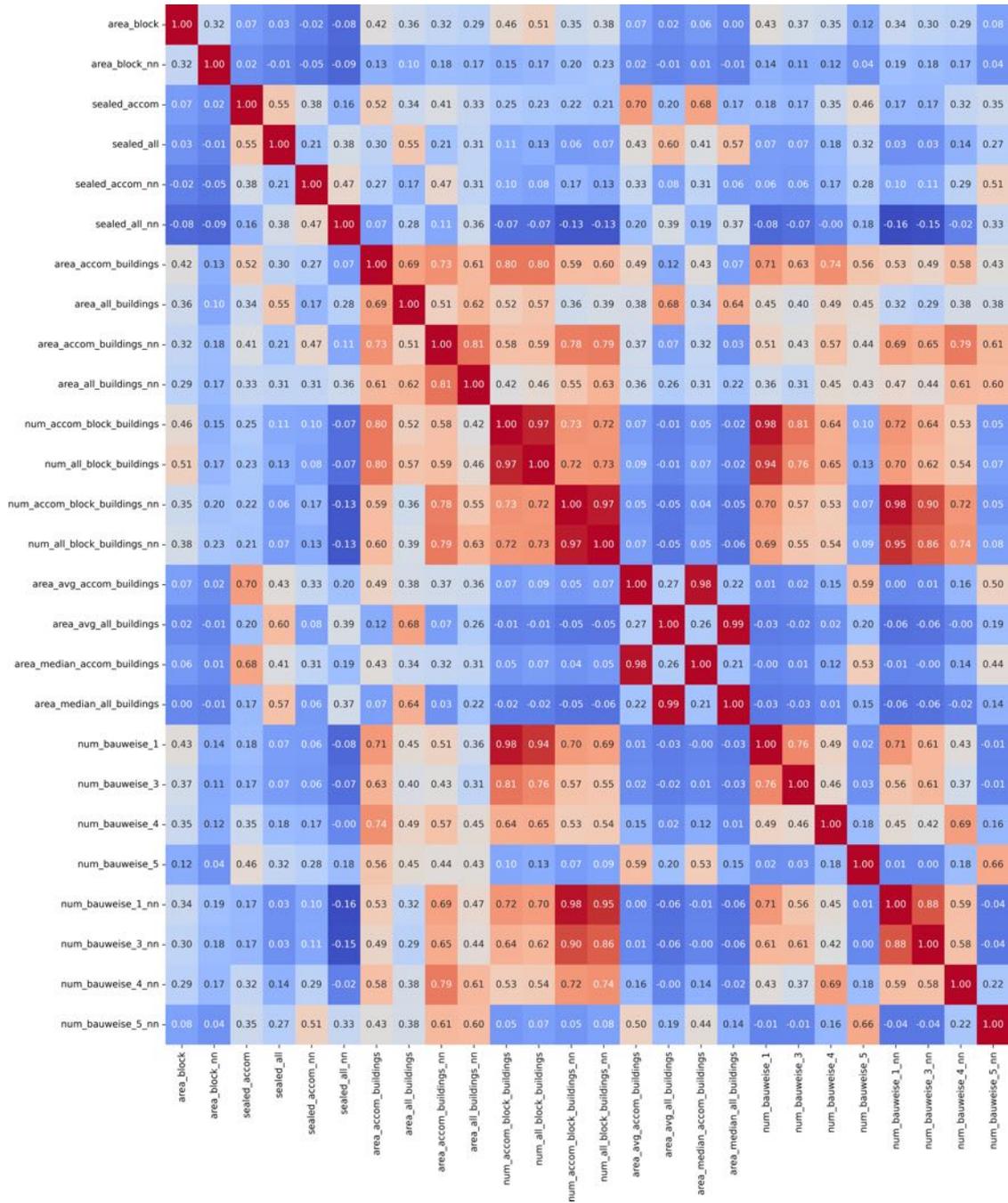


Abbildung B.10: Vollständige Korrelationsmatrix aller 26 Features für K-Means

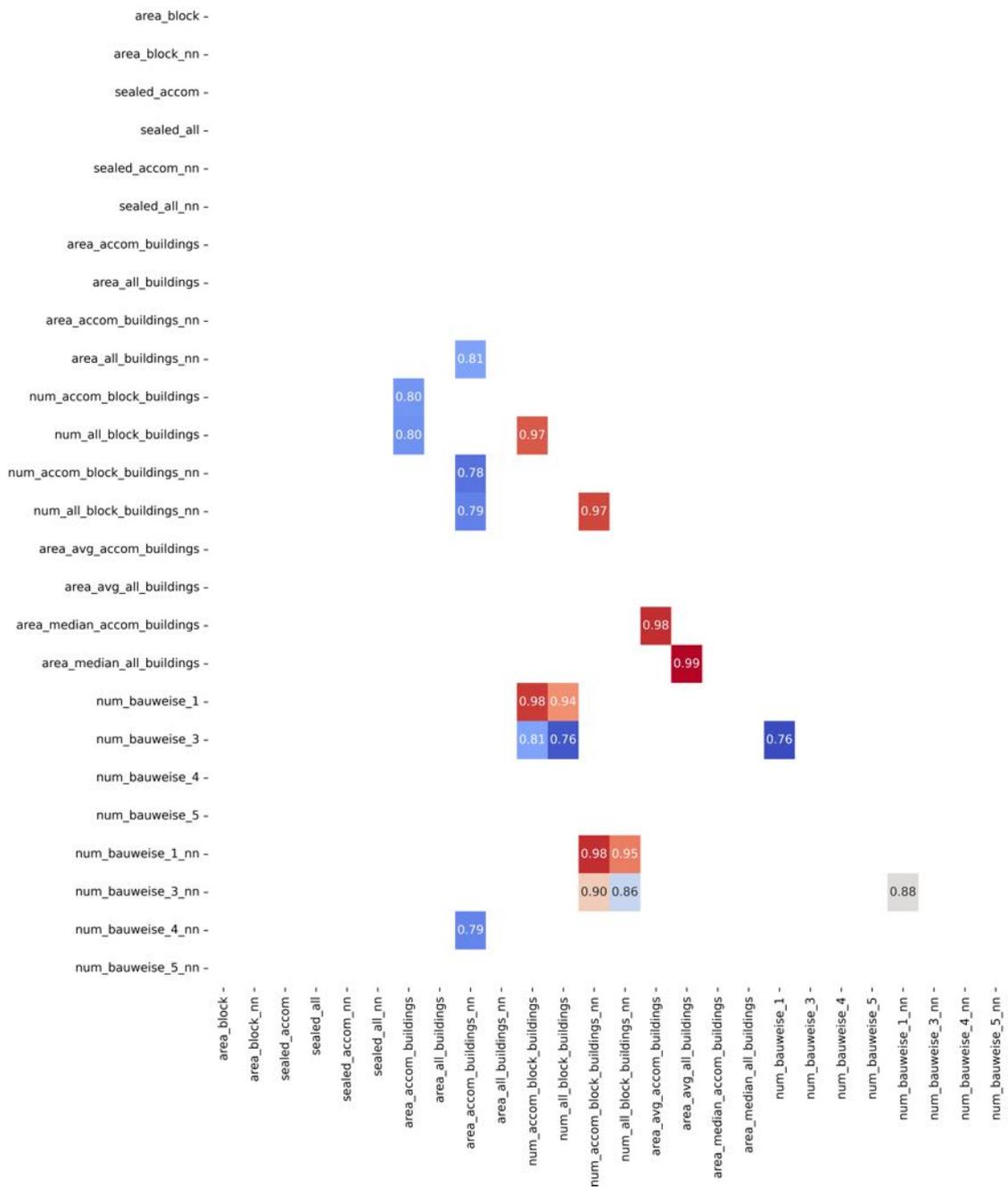
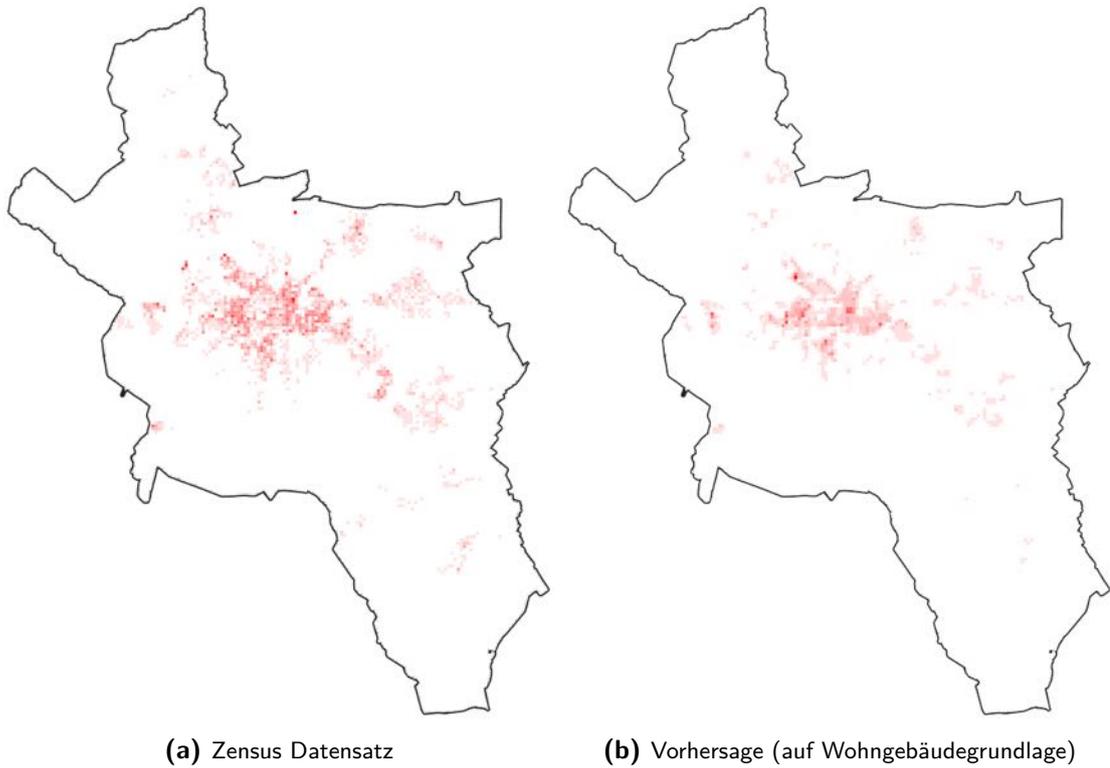
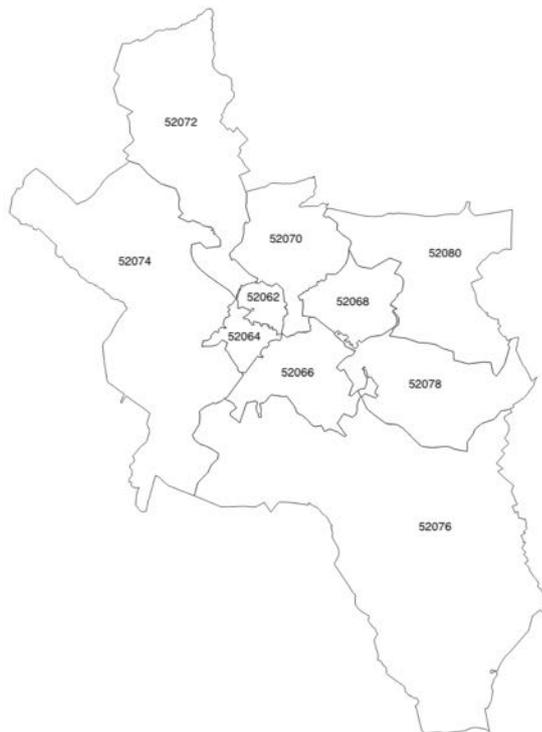


Abbildung B.11: Korrelationsmatrix mit stark korrelierenden Features für K-Means



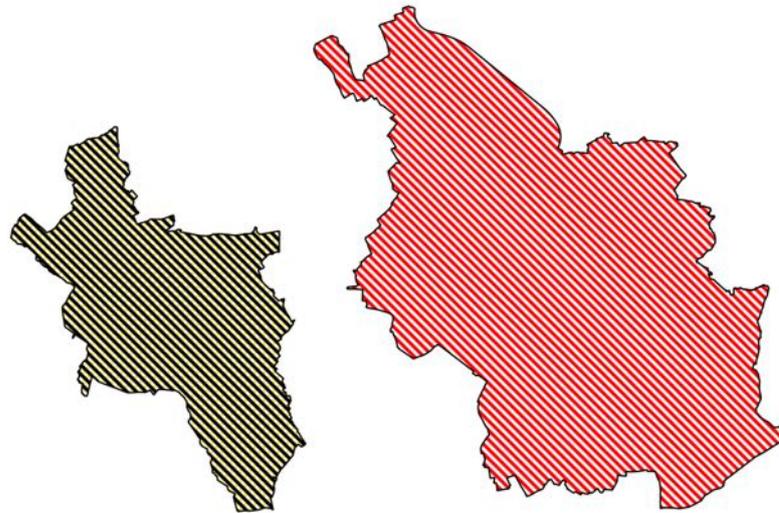
**Abbildung B.12:** Zensus-Verteilung und eigene Vorhersage der Einwohnerzahlen im  $100\text{ m} \times 100\text{ m}$  Gitter in Aachen. Umso roter das Gitterfeld, umso mehr Einwohner wohnen in diesem Bereich. Das Maximum liegt bei 500



**Abbildung B.13:** Postleitzahlgebiete im Stadtbezirk Aachen

## C Exemplarischer Verfahrensablauf

Der gesamte Ablauf wird an einem konkreten Beispiel vorgestellt. Als zu analysierende AOI wird die Verwaltungsgrenze der Stadt Aachen mit einer Fläche von  $160,8 \text{ km}^2$  und als Trainings-AOI die Verwaltungsgrenze der Stadt Köln mit einer Fläche von  $405,2 \text{ km}^2$  genutzt. In Abbildung C.1 sind die verwendeten Polygone der beiden AOIs abgebildet. Das vorgestellte Beispiel beschränkt sich im Training der NNs auf die folgenden OOs: Gesamtgebäude, Wohngebäude, freistehende Gebäude, Doppelhaushälften und gereichte Häuser.



**Abbildung C.1:** Abbildungen zu Aachen als Analyseregion und Köln als Trainingsregion  
Aachen (links) als Analyse-AOI und Köln (rechts) als Trainings-AOI

Zunächst beginnt die Phase der Trainingsvorbereitung in welcher als erstes die Trainingsdaten heruntergeladen werden. Da sich die Trainingsregion komplett im Bundesland NRW befindet, wird ausschließlich dessen WMS angesprochen. Die Bilder werden als GeoTiffs im RGB-Layer in EPSG:3857 mit  $600 \times 600$  Pixeln ausgeliefert, sodass keine Umwandlung des Dateiformates oder eine manuelle Georeferenzierung der NRW-Bilder nötig ist. Als Zoomlevel wurde  $z = 17$  gewählt und für die Region Köln insgesamt 11315 Bildkacheln heruntergeladen. Diejenigen Bildkacheln, die auf der Grenze des Kölner Polygons liegen, werden im Anschluß zugeschnitten. Das Herunterladen der Bildkacheln muss nicht für jedes der fünf OOs in der Region Köln erneut getätigt werden, da diese im Speicher durch eine eindeutige Kachelnummerierung gecached werden können.

Für die jeweiligen OOs wird der Datenbestand der gesamten Region Köln in einem Aufruf heruntergeladen. Die Gesamtmenge an OOI-Polygonen zum Labelling wird wie die Bilder in 11315 Kacheln unterteilt. Die Koordinaten für diese Kacheln liefert die Bounding Box, die in den GeoTiffs kodiert ist. Damit ist das Erstellen der Feature-Label-Paare für die Trainingsregion abgeschlossen. Die Paare werden in zwei disjunkte Mengen mit 9052 und 2263 Kacheln unterteilt. Erstere dient zum Training des NN und zweitere zur ersten Evaluierung innerhalb

der Region Köln.

Für das Training der neuronalen Netze auf die fünf OOIs mit RasterVision wurden die folgenden Einstellungen verwendet:

- **Epochs:** 5 ( $\Rightarrow 5 \cdot 9052 = 45260$  Bilder wurden in jedes NN zum Training eingespeist)
- **Batchgröße:** 8 ( $\Rightarrow$  nach 8 Bildern wurden stets Gewichtungen im NN angepasst)
- **Learning-Rate:** 0,0001
- **Backbone:** ResNet50

Zur Durchführung wurde ein Hochleistungsrechner am High Performance Computing Cluster der RWTH verwendet. Dafür wurde die folgende Hardware genutzt:

- **CPUs:** 8 Kerne x Intel Platinum 8160 mit 2,1 GHz Grundtaktung
- **GPUs:** 1 x Nvidia Tesla V100
- **RAM:** 4 GB pro CPU  $\Rightarrow$  32 GB insgesamt

Das Training eines NN dauerte je OOI zwischen 7 und 7,5 Stunden. Dabei betrug die Durchführungsdauer einer Epoch jeweils 40 bis 45 Minuten, sodass das eigentliche Training bereits unter 4 Stunden abgeschlossen war. Die weiteren 3 Stunden wurden zur Berechnung der ersten Ergebnisse anhand der 2263 Evaluierungskacheln in der Region Köln verwendet.

Nach der abgeschlossenen Trainingsdurchführung wurde das generierte Model-Bundle genutzt, um in der zu analysierenden AOI Vorhersagen zu tätigen. Für die Vorhersagen wurden zunächst 4559 Bildkacheln der Region Aachen heruntergeladen. Die Vorhersagen haben dieselbe CPU-Hardware genutzt wie das Training. Eine GPU wurde für eine Vorhersage nicht mehr benötigt. Durch Voruntersuchungen konnte nämlich herausgefunden werden, dass eine GPU-Nutzung die Vorhersagephase nicht verkürzt.

Die Vorhersage auf alle Aachener Bildkacheln dauerte fast 8 Stunden pro OOI-Durchlauf. Die vorhergesagten 4559 Polygonkacheln waren danach in einer PostGIS Datenbank unter der Tabelle *predicted\_building* mit der entsprechenden Experiment-ID abrufbar. Die PostGIS Datenbank lief auf einem virtuellen Server mit 16 vCPUs und 32GB RAM. Im nächsten Schritt wurde ein SQL-Script zum Postprocessing der Bildkacheln gestartet, welches alle vorhergesagten Aachener Polygone in unter 3 Minuten zusammenführte und nachbearbeitete.

Die zusammengeführten Polygone waren in *cached\_predicted\_building* geladen worden. In der Tabelle *cached\_osm\_building* wurden zuvor die OSM Polygone und ALKIS Bauweisen der Gebäude in Aachen hochgeladen. Mit den Daten in den beiden *cached*-Tabellen konnten nun die approximierten Metriken für die Vorhersagen in Aachen berechnet werden. Die Berechnung der Metriken benötigte knapp 5 Minuten je OOI.

Damit ist der erste Teil in unter 16 Stunden abgeschlossen und alle benötigten Polygone sind für den zweiten Teil berechnet worden. Ein Ergebnis der NN für alle Gebäude (*Buildings*) und ausschließlich bewohnte Gebäude (*Accommodation*) ist in Abbildung C.2 zu sehen:

Im ersten Schritt werden in der zweiten Phase die drei Bauweisen-OOIs (freistehende Gebäude,



**Abbildung C.2:** Vorhergesagte Gesamtgebäude (blau) und Wohngebäude (gelb im Vordergrund)

Doppelhaushälften und gereichte Häuser) zur Vorklassifizierung der bewohnten Gebäudepolygone (*Accommodation*) genutzt. Das Ergebnis der Vorklassifizierung der Aachener Innenstadt ist in Abbildung C.3 zu sehen.



**Abbildung C.3:** Vorklassifizierung der Aachener Wohngebäude

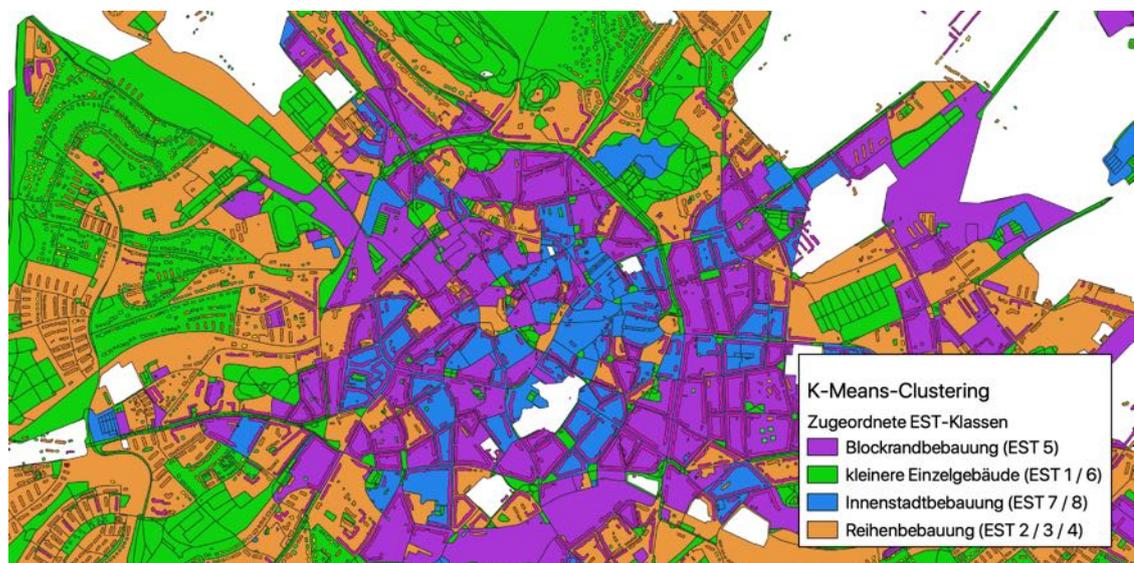
Die noch nicht klassifizierten Gebäude erhalten ihre Bauweise durch ein RandomForest-Modell. Dieses wurde auf die vorherigen drei Bauweisen und zusätzlich Blockrandbebauungen in der Region Köln trainiert. Um das Training realistischer zu machen, wurden dafür nicht die perfekten Polygone genutzt, sondern die Informationen auf in Köln vorhergesagte Gebäude übertragen. Dafür wurde das in Köln trainierte NN für Wohngebäude zur kompletten Vorhersage auf die eigene Trainingsregion angewendet. Der mit Kölner Vorhersagen trainierte RandomForest-Klassifikator wird dann auf die Region Aachen angewendet. Bei einer Zuweisung der Bauweise Blockrandbebauungen durch RandomForest wird die Vorklassifizierung

überschrieben, da diese den Gebäudetyp Blockrandbebauung nicht kannte. Ansonsten wird die Vorklassifizierung bevorzugt. Das Resultat ist in Abbildung C.4 zu sehen.



**Abbildung C.4:** Alle vorhergesagten Wohngebäude in Aachen wurde einer Bauweise zugewiesen

Nachdem die Bautypen zugewiesen wurden, kann die Zuordnung der EST-Klassen der Baublöcke mit dem K-Means-Clustering stattfinden. Da es sich bei K-Means um ein Unsupervised-Learning-Verfahren handelt, wird der Algorithmus mit  $k = 4$  als Eingabeparameter direkt auf die dafür berechneten Features in der Region Aachen angewendet. Die Zuordnung ist der Abbildung C.5 zu entnehmen.



**Abbildung C.5:** Anwendung des K-Means-Clustering auf die künstlich generierten Baublöcke

Im letzten Schritt kann nun die Zuordnung des Energiebedarfes getätigt werden. Die Zuordnung lässt sich auf zwei Grundflächenarten anwenden: Gesamtgebäude oder Wohngebäude (siehe dafür auch Flächen für Buildings und Accommodation in Abbildung C.2). Die dazugehörigen Ergebnisse der beiden Flächen wurden in Kapitel 4 bewertet.



und `building` für Gebäude im Allgemeinen) auf Korrelation untereinander verglichen. Der Auszug der reinen Gegenüberstellung von `building`-Features zueinander wurde weggelassen, da sich die Ergebnisse fast identisch mit dem internen `accom` Vergleich decken.

Im Folgenden wird auf einige Korrelationen der berechneten Merkmale kurz eingegangen. Die beiden stärksten Korrelationen in Abbildung D.1a gibt es zwischen den Features `area/perimeter` und `max_diameter/rect_length` (gelesen X/Y-Achse ohne den Präfix `accom`). Dies erscheint auch logisch, da eine größere Fläche mit einem größeren Umfang einhergeht und der Durchmesser des kleinsten umliegenden Kreises gleich der Länge des minimalen Rechtecks. Ein größeres Cluster an Korrelationen lässt sich zwischen den Rechteck-Eigenschaften (`rect_area`, `rect_width` und `rect_length`) und den Eigenschaften des ursprünglichen Polygons (`area` und `perimeter`) festmachen. Auch hier lässt sich dies damit begründen, dass die jeweiligen Flächenmerkmale sich zum Großteil aus den anderen flächenbezogenen Features ableiten lassen. Generell lässt sich diese Begründung für fast alle weiteren Korrelationen übernehmen. Einzig die Korrelation zwischen der Blockgröße (`urban_block_buildings_area`) und der Anzahl der Gebäude je Block (`num_urban_block_buildings`) besteht nicht zwischen zwei reinen Flächenmerkmalen. Die Korrelation lässt sich aber auch hier ähnlich leicht begründen. Durch eine größere Blockgröße ist es wahrscheinlicher ein dünner besiedeltes Gebiet zu erlangen. In solchen Gebieten sind EFH/MFH häufiger, sodass die Anzahl der Gebäude ebenfalls zunimmt. Letztendlich lassen sich folgende Features streichen, wobei angegeben wird welches andere Feature bestehen bleibt, da dieses durch eine starke Korrelation die Informationen substituiert:

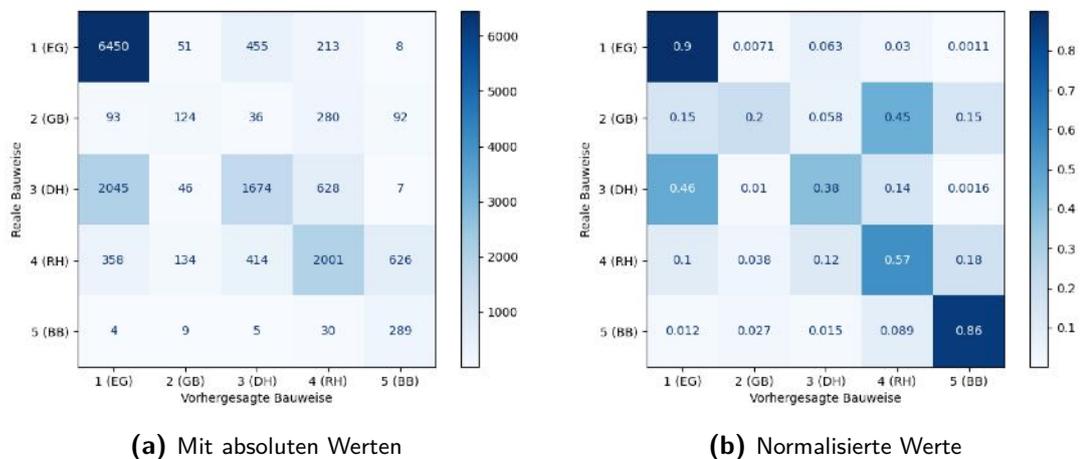
- $\Rightarrow$  := wird substituiert durch
- `perimeter`, `max_diam`, `rect_area`, `rect_width` und `rect_length`  $\Rightarrow$  `area`
- `area_ratio`  $\Rightarrow$  `convex`
- `urban_block_buildings_area`  $\Rightarrow$  `num_urban_block_buildings`

Die Abbildung D.1b stellt außerdem dar, dass es sinnvoll war das Feature-Set über zwei verschiedene Polygongrundlagen für das selbe Gebäude errechnen zu lassen. Schließlich korrelieren die `accom`-Merkmale kaum mit den `building`-Merkmalen, sodass neue Informationen für die spezifische Eigenschaften von Gebäudetypen gewonnen werden können. Die stärkste Korrelation besteht zwischen `buildings_num_urban_block_buildings` und `accom_num_urban_block_buildings` mit 0,95. Ersteres Feature zählt alle erkannten vollständigen Gebäudepolygone innerhalb eines Blocks. Zweiteres zählt die Anzahl der reinen Wohngebäudepolygone. Eine hohe Korrelation scheint also logisch, da viele Gebäude in Aachen Wohneinheiten beinhalten. Die weiteren Korrelationen lassen sich mit der gleichen Begründung, wie im vorherigen Absatz erklären. Zuletzt wird nochmal aufgelistet, welche Features an dieser Stelle, wie substituiert werden:

- `buildings_num_building_buffer`  $\Rightarrow$  `accom_num_rings`
- `buildings_num_urban_block_buildings`  $\Rightarrow$  `accom_num_urban_block_buildings`

Nachdem die stark korrelierenden Features untereinander gefiltert wurden, wurde betrachtet, welche einzelnen Features keine Informationen bringen. Dabei sticht das Merkmal `building_angle` in der vollständigen Korrelationsmatrix ins Auge. Dieses korreliert mit keinem anderen Merkmal (überall Werte nahe 0), sodass die Vermutung nahe liegt, dass es sich hierbei um eine Zufallsverteilung zwischen Gebäudetyp und -winkel handelt. Daher wird das Feature in der `accom` und `building` Ausprägung von den RandomForest-Berechnungen ausgeschlossen. Damit ist die Auswahl eines reduzierten Feature-Sets für das Training abgeschlossen.

Nach einem Training des RandomForest-Klassifikators mit allen 5 Bauweisen in der üblich verwendeten Region Köln und der Anwendung in der Region Aachen ergeben sich die Konfusionsmatrizen in Abbildung D.2.



**Abbildung D.2:** Konfusionsmatrizen für Random-Forest-Klassifikation in 5 Bauweisen

Die linke Matrix zeigt die Gebäudezuordnung in absoluten Zahlen und die rechte Matrix normalisiert die Werte, indem die Felder durch die Summe der jeweiligen Zeile geteilt werden. Auf der Hauptdiagonalen wird somit der *accuracy*-Score der jeweiligen Bauweise ausgewertet. In der absoluten Verteilung fällt auf, dass es überwiegend Häuser der Bauweisen 1, 3, und 4 gibt in Aachen. Die Bauweisen 2 und 5 sind deutlich weniger vertreten, nämlich deutlich unter 1000 Gebäude je Klasse. In der rechten Matrix lässt sich jedoch erkennen, dass trotz der wenigen Vorkommen der Bauweise 5 die Zuweisung in 86% der Fälle korrekt stattgefunden. Bei Bauweise 2 wurden nur 20% richtig zugeordnet. Ein Großteil mit 45% der echten Bauweise 2 wurde fälschlicherweise der Bauweise 4 zugewiesen. Eine Erklärung könnte lauten, dass beide Bauweisen in der Ebene sehr ähnlich aussehen.

## E Voruntersuchungen zum K-Means-Clustering

Für die Zuordnung der EST-Klassen zu den Baublöcken existiert kein Referenzdatensatz, sodass keine Aussage über die finale Verteilung getroffen werden kann. Jedoch lässt sich genau wie für die RandomForest-Klassifikation die Korrelationsmatrix auswerten, sodass weniger aussagekräftige Features vor Anwendung des K-Means-Clustering gefiltert werden können. Die vollständige Korrelationsmatrix zu allen potentiellen Features für das K-Means-Clustering (siehe Tabelle A.5) ist mit Abbildung B.10 angehängen. Zur übersichtlicheren Untersuchung wurde mit Abbildung B.11 die Korrelationsmatrix mit den stark korrelierenden Werten verwendet. Das Vorgehen zum Streichen von Features ist dasselbe wie zuvor bei den RandomForest Untersuchungen, sodass auf eine detaillierte Erklärung verzichtet wird.

Neben der paarweisen Filterung der stark korrelierenden Features werden weitere Features aussortiert, die eine geringe Aussagekraft haben. Dies betrifft zunächst die beiden Features, die sich auf die Flächengröße der Blöcke beziehen (`area_block` und `area_block_nn`). Schließlich hatte sich bereits in den DecisionTree-Versuchen (siehe Unterabschnitt 3.2.3) in der Methodik ergeben, dass die Größe der künstlichen Baublöcke keine Aussage über den Stadtraumtyp trifft. Außerdem wurden alle vier Features gestrichen, die die Grundflächen der Gebäude innerhalb eines Blocks aufaddieren (`area_accom_buildings`, `area_all_buildings`, `area_accom_buildings_nn` und `area_all_buildings_nn`). Im Gegensatz dazu sind die Features zur Versiegelung (Präfix `sealed`) aussagekräftiger, da diese die Blockgröße und die Bebauung im Block ins Verhältnis setzen. Nach einigen Versuchen hat sich ergeben, dass es sinnvoll ist die Features mit der Anzahl der jeweiligen Bauweisen, die auf den eigenen Block bezogen sind, zu behalten und die der Nachbarblöcke im Feature-Set zu streichen.

Am Ende wurden für das Clustering die folgenden 10 Features genutzt, um die Baublöcke in die 4 zusammengefassten EST-Klassen, die in der Methodik vorgestellt wurden, zu unterteilen:

- $\Rightarrow$  := wird substituiert durch
- Merkmale, die auf die bauliche Versiegelung des Blocks und der Nachbarblöcke bezogen sind: `sealed_accom`, `sealed_all`, `sealed_accom_nn` und `sealed_all_nn`
- Die durchschnittliche Gebäudegrundfläche der Gebäude und Wohngebäude im Block: `area_avg_accom_buildings` und `area_avg_all_buildings`
- Anzahl der jeweiligen 4 Bauweisen, die durch Vorklassifizierung und RandomForest erkannt wurden, innerhalb des Blocks: `num_bauweise_1`, `num_bauweise_3`, `num_bauweise_4` und `num_bauweise_5`