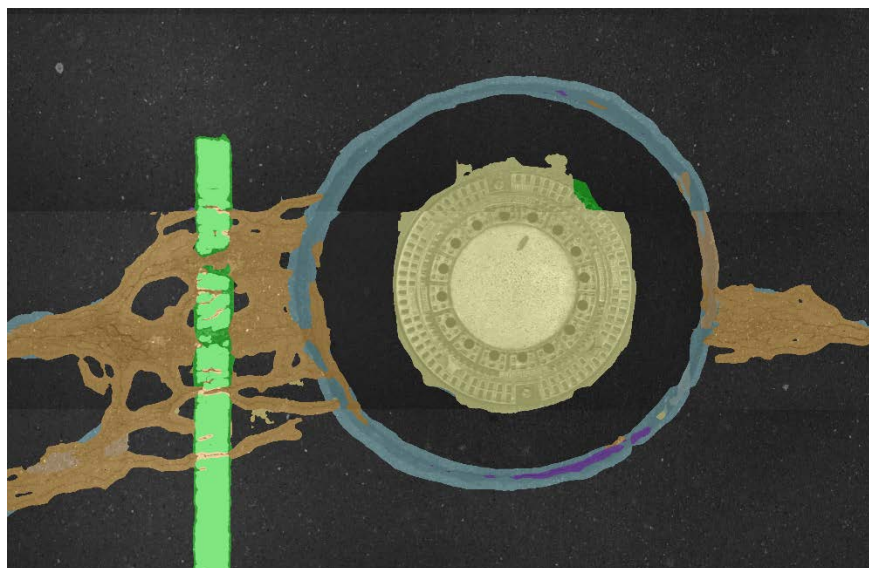


Automatisierte Schadstellenerkennung für unterschiedliche Fahrbahnbeläge mittels Deep-Learning-Techniken

ASFaLT

Ein Projekt finanziert im Rahmen der
D-A-CH Kooperation
Verkehrsinfrastrukturforschung 2018

September 2021



Impressum:

Herausgeber und Programmverantwortung:

Bundesministerium für Verkehr und digitale Infrastruktur (BMVI)
Invalidenstraße 44
10115 Berlin
Deutschland




Bundesministerium
für Verkehr und
digitale Infrastruktur

Bundesministerium für Klimaschutz (BMK)
Radetzkystraße 2
1030 Wien
Österreich



Bundesministerium
Klimaschutz, Umwelt,
Energie, Mobilität,
Innovation und Technologie

Bundesamt für Strassen (ASTRA)
Mühlestrasse 2, Ittigen
3003 Bern
Schweiz



Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Bundesamt für Strassen ASTRA

Für den Inhalt verantwortlich:

TU Ilmenau
- Institut für technische Informatik und Ingenieurinformatik
Am Ehrenberg
98693 Ilmenau
Deutschland



TECHNISCHE UNIVERSITÄT
ILMENAU

Programmmanagement:

Österreichische Forschungsförderungsgesellschaft mbH
Thematische Programme
Sensengasse 1
1090 Wien
Österreich



FFG
Forschung wirkt.

Automatisierte Schadstellenerkennung für unterschiedliche Fahrbahnbeläge mittels Deep-Learning-Techniken

ASFaLT

Ein Projekt finanziert im Rahmen der
D-A-CH Kooperation
Verkehrsinfrastrukturforschung 2018

AutorInnen:

Ronny STRICKER

Dustin AGANIAN

Johann LEMBACH

Benedict STEPHAN

Maximilian SESSELMANN

Thorsten NABER

Roland SPIELHOFER

Matthias HAHN

Astrid HAUTZ

Auftraggeber:

Bundesministerium für Verkehr und digitale Infrastruktur, Deutschland Bundesministerium für Verkehr, Innovation und Technologie, Österreich Bundesamt für Strassen, Schweiz

Auftragnehmer:

TU Ilmenau - Institut für technische Informatik und Ingenieurinformatik
Am Ehrenberg
98693 Ilmenau
Deutschland

AIT Austrian Institute of Technology GmbH
Giefinggasse 4
1210 Wien
Österreich

LEHMANN + PARTNER GmbH
Schwerborner Straße 1
99086 Erfurt
Deutschland

VIA IMC GmbH
Franz-Ehrlich-Straße 5
12489 Berlin
Deutschland

Inhalt

| | | |
|-------|--|----|
| 1 | Stand des Wissens und Zielsetzung | 10 |
| 1.1 | Automatisierte Detektion von Schäden an der Straßeninfrastruktur | 10 |
| 1.2 | Modellierung von Unsicherheiten in Neuronalen Netzwerken | 11 |
| 1.3 | Synchronisierung multitemporaler Datensätze / Schadensentwicklung..... | 11 |
| 1.4 | Zielsetzung | 12 |
| 1.5 | Vorgehensweise | 13 |
| 2 | Katalogisierung von relevanten Schadensklassen im D-A-CH-Raum..... | 15 |
| 3 | Datengrundlage für die automatisierte Detektion | 18 |
| 3.1 | Definition von Labelklassen | 18 |
| 3.2 | Labeling von Oberflächenbildern | 19 |
| 3.3 | Datenbasis..... | 22 |
| 3.3.1 | Adaption von Betonklassen | 24 |
| 3.3.2 | Klassenmappings | 26 |
| 4 | Automatisierte Schadstellendetektion | 27 |
| 4.1 | Eingesetzte Segmentierungsverfahren | 27 |
| 4.1.1 | U-Net..... | 27 |
| 4.1.2 | Typische aktuelle Segmentierungsverfahren | 29 |
| 4.1.3 | Verwendete Fehlerfunktionen | 32 |
| 4.2 | Bewertungsmaße..... | 34 |
| 4.2.1 | Mean Intersection over Union | 34 |
| 4.2.2 | Occurrence Score..... | 34 |
| 4.3 | Anpassung zur Berücksichtigung von erweitertem Kontext..... | 36 |
| 4.3.1 | Positionskarte..... | 37 |
| 4.3.2 | Bildnahtkarte..... | 37 |
| 4.3.3 | Ergebnisse..... | 38 |

| | | |
|-------|---|----|
| 4.4 | Segmentierungsergebnisse | 39 |
| 4.4.1 | Vorgehensweise bei der Visualisierung der Ergebnisse auf Asphalt Daten .. | 39 |
| 4.4.2 | Voruntersuchungen auf alten Datensätzen | 40 |
| 4.4.3 | Voruntersuchungen auf dem Datensatz „2020_06 AIT & L+P“ | 41 |
| 4.4.4 | Modell- und Parameteroptimierungen auf dem Datensatz „2020_06 AIT & L+P“ | 42 |
| 4.4.5 | Visualisierung der Ergebnisse auf Asphalt Daten | 43 |
| 4.5 | Segmentierungsergebnisse auf Beton | 48 |
| 4.5.1 | Quantitative Bewertung..... | 48 |
| 4.5.2 | Qualitative Bewertung..... | 49 |
| 4.6 | Vergleich mit menschlicher Segmentierungsleistung | 52 |
| 4.6.1 | Methodik..... | 52 |
| 4.6.2 | Ergebnisse..... | 53 |
| 4.7 | Fazit..... | 53 |
| 5 | Qualitätsmaß für die Detektionsgenauigkeit..... | 54 |
| 5.1 | Arten von Unsicherheiten..... | 54 |
| 5.2 | Methoden zur Bestimmung von Unsicherheiten | 55 |
| 5.2.1 | Bayes'scher Ansatz | 56 |
| 5.2.2 | Approximierende bayessche Ansätze | 56 |
| 5.2.3 | Nicht-bayessche Ansätze | 58 |
| 5.2.4 | Fazit..... | 58 |
| 5.3 | Anwendung auf synthetischen Daten | 59 |
| 5.3.1 | Datengrundlage | 59 |
| 5.3.2 | Netzwerkmodelle | 60 |
| 5.3.3 | Erkennung von unbekanntem Daten..... | 61 |
| 5.3.4 | Schätzung der Datenunsicherheit..... | 63 |
| 5.4 | Unsicherheiten bei der Anwendung auf Straßendaten | 64 |
| 5.4.1 | Erkennung von unbekanntem Daten..... | 64 |

| | | |
|-------|---|----|
| 5.4.2 | Modellunsicherheit..... | 65 |
| 5.5 | Zwischenfazit..... | 68 |
| 5.6 | Modellunsicherheiten für Segmentierungsansätze..... | 68 |
| 5.6.1 | Unsicherheiten an Klassengrenzen | 70 |
| 5.6.2 | Bewertung der Modellunsicherheit..... | 71 |
| 5.7 | Selektion von Bildern mit großer Unsicherheit für das Labeling | 71 |
| 5.8 | Einsatz von Unsicherheiten für die Bewertung von Messfahrten..... | 72 |
| 6 | Unterscheidung unterschiedlicher Straßenoberflächen | 73 |
| 6.1 | Datengrundlage | 74 |
| 6.2 | Unterscheidung von Beton und Asphalt..... | 74 |
| 6.2.1 | Normalisierung von Bildpatches..... | 75 |
| 6.2.2 | Ergebnisse..... | 76 |
| 6.3 | Unterscheidung unterschiedlicher Ausprägungen von Oberflächenarten | 77 |
| 6.3.1 | Datengrundlage | 77 |
| 6.3.2 | Training | 78 |
| 6.3.3 | Ergebnisse..... | 79 |
| 7 | Abbildung der Automatisierten Detektion auf die länderspezifischen Bewertungen...80 | |
| 7.1 | Abbildung von Detektionsklassen in Schadensklassen..... | 81 |
| 7.2 | Nachverarbeitung der Detektionsergebnisse | 81 |
| 7.3 | Umwandlung von Flickstellen..... | 83 |
| 7.3.1 | Bestimmung der Fläche über den Umriss | 84 |
| 7.3.2 | Bestimmung der Fläche über die Oberflächenstruktur | 84 |
| 7.3.3 | Bestimmung von Flächenmaßen | 85 |
| 7.3.4 | Bestimmung von Längenmaßen | 85 |
| 7.3.5 | Detektion von Netzzissen..... | 86 |
| 7.3.6 | Lokalisation von Schäden auf Betonplatten | 86 |
| 8 | Demonstrator | 87 |

| | | |
|--------|---|-----|
| 8.1 | Rahmenbedingungen und technische Anforderungen..... | 88 |
| 8.2 | Benutzeroberfläche..... | 88 |
| 8.2.1 | Messfahrtenübersicht | 88 |
| 8.2.2 | Datenimport..... | 89 |
| 8.2.3 | Messfahrtenanalyse..... | 90 |
| 8.2.4 | Auswertungsvergleich..... | 91 |
| 8.3 | Umsetzung der User-Stories..... | 92 |
| 8.3.1 | Basic User-Stories | 92 |
| 8.3.2 | Nice-to-have User-Stories..... | 95 |
| 8.4 | Benutzertests..... | 96 |
| 9 | Vergleich mit traditioneller standardisierter Bewertung..... | 97 |
| 9.1 | Datengrundlage | 97 |
| 9.1.1 | Testdaten aus Deutschland | 98 |
| 9.1.2 | Testdaten aus Österreich..... | 99 |
| 9.2 | Auswertung nach ZEB-Standard..... | 99 |
| 9.2.1 | Asphalt | 100 |
| 9.2.2 | Beton..... | 105 |
| 9.2.3 | Fazit..... | 107 |
| 9.3 | Auswertung nach RVS-Standard | 108 |
| 9.3.1 | Fazit..... | 113 |
| 9.4 | Laufzeit..... | 114 |
| 9.5 | Fazit..... | 114 |
| 10 | Bewertung der Schadensentwicklung..... | 115 |
| 10.1 | Räumliche Synchronisation von Straßenbildern..... | 115 |
| 10.1.1 | Räumliche Synchronisation von Oberflächenbildern aus zeitlich versetzten Aufnahmen | 115 |
| 10.1.2 | Räumliche Synchronisation von Oberflächen- und Frontbildern..... | 122 |
| 10.1.3 | Fazit..... | 125 |

| | | |
|--------|--|-----|
| 10.2 | Räumliche Synchronisation von Schadstellen..... | 125 |
| 10.2.1 | Konzept | 126 |
| 10.3 | Analyse der Schadensentwicklung..... | 127 |
| 10.3.1 | Experimentelle Analyse und Evaluierung..... | 129 |
| 10.3.2 | Fazit..... | 134 |
| 11 | Übertragbarkeit auf andere Anwendungsdomänen..... | 134 |
| 11.1 | Anwendungsdomäne und Datenlage | 134 |
| 11.2 | Experimente und Ergebnisse der Machbarkeitsstudie..... | 137 |
| 12 | Fazit..... | 141 |
| 13 | Literaturverzeichnis | 144 |
| 14 | Anhang..... | 150 |
| 14.1 | Anhang A..... | 150 |

1 Stand des Wissens und Zielsetzung

Eine vollständige Automatisierung bezüglich der Erkennung von Straßenschäden bei der Auswertung von Oberflächenbildern wurde bisher von noch keinem Verfahren erreicht. Es gibt jedoch erste Ansätze, die im Rahmen dieses Vorhabens zu lösende Teilprobleme behandeln. Der Stand der Technik zu diesen Teilproblemen wird nachfolgend thematisch geordnet beschrieben:

1.1 *Automatisierte Detektion von Schäden an der Straßeninfrastruktur*

Die aus der Literatur bekannten Verfahren zur automatisierten Detektion von Schadstellen auf Straßendaten können grob in zwei Gruppen eingeteilt werden.

Die erste Gruppe von Verfahren geht von der Annahme aus, dass sich Risse als lokale Intensitätsminima identifizieren lassen. Entsprechend werden Schadstellen über eine Schwellwertoperation extrahiert. Für die weitere Bearbeitung der dabei gefundenen Schadstellenkandidaten kommen beispielsweise morphologische Operatoren [Peng 2015, Oliveira 2014, Xu 2013, Chambon 2011] oder graphen-basierte Analysen [Oliveira 2017, Amhaz 2016, Tang 2013, Zou 2012] zum Einsatz.

Im Gegensatz zu den erstgenannten Verfahren, basiert die zweite Gruppe von Verfahren auf der Auswertung von Bildausschnitten durch unterschiedliche Klassifikatoren. Dabei kommen zum einen klassische, von einem Designer vordefinierte Merkmalsextraktionsverfahren wie Histogram of Oriented Gradients (HOG) [Kapela 2015] oder Local Binary Patterns (LBP) [Quintana 2016, Varadharajan 2014] in Kombination mit Support Vektor Maschinen als Klassifikator zum Einsatz. Zum anderen werden Convolutional Neural Networks für die Detektion von Schäden eingesetzt [Fan 2018, Eisenbach 2017, Pauly 2017, Wang 2017], die die problemspezifischen Merkmalsdetektoren im Ergebnis eines datengetriebenen Lernprozesses selbständig herausbilden. Wie im Rahmen des ASINVOS Projektes in [Eisenbach 2017] bereits gezeigt werden konnte, sind klassische schwellwertbasierte Ansätze als auch Verfahren mit klassischen, designten Merkmalsextraktionsverfahren bei der Detektion den auf Convolutional Neural Networks basierenden Ansätzen deutlich unterlegen.

1.2 **Modellierung von Unsicherheiten in Neuronalen Netzwerken**

Die für Klassifikationsaufgaben überwiegend eingesetzte Softmax-Ausgabefunktion von Neuronalen Netzwerken normiert die Ausgaben für die einzelnen Klassen auf das Intervall $[0, \dots, 1]$. Da auch die Summe der Ausgaben über alle Klassen auf 1 normiert ist, wird die Softmax-Ausgabe oft fälschlicherweise als Qualitätsmaß für die Sicherheit der Entscheidung eines Neuronalen Netzwerkes interpretiert, z.B. in [Gopalakrishnan 2017]. Dies ist jedoch nicht der Fall, da für die gelernten Gewichte eines Neuronalen Netzwerkes nur eine Punktschätzung vorliegt, d.h. die Varianzen bezüglich der Sicherheit in der Entscheidung bleiben unberücksichtigt [Gal 2016]. Um dieses Problem zu beheben, muss die Unsicherheit explizit in das Neuronale Netzwerk eintrainiert werden. Die Art und Weise dies zu realisieren ist aktueller Forschungsgegenstand. Eine Vielfalt von möglichen Lösungsansätzen wurden auf den „Bayesian Deep Learning“-Workshops der NIPS-Konferenzen 2016 und 2017 vorgestellt. Vielversprechende Ansätze sind Bayessche Inferenz [Kendall 2017], Ensemblestrategien [Lakshminarayanan 2017], Kalibrierung der Netzwerkausgabe [Guo 2017] und eine spezifische Netzwerkausgabecodierung [Kardan 2017]. Bisher hat sich noch keiner der Ansätze als der beste herausgestellt.

In der Anwendungsdomäne der Straßenschadenerkennung wurde in [Gopalakrishnan 2017] die Softmax-Ausgabefunktion als vermeintliches Maß der Unsicherheit verwendet. Dieses ist nach aktueller Literaturlage aus den o.g. Gründen jedoch ungeeignet. In eigenen Vorarbeiten [Seichter 2018] im Rahmen des Projekts ASINVOS wurde die Modellierung der Unsicherheit mittels Bayesscher Inferenz erfolgreich eingesetzt, um gezielt Trainingsdaten auszuwählen, die dem Klassifikator helfen, besser zu generalisieren.

1.3 **Synchronisierung multitemporaler Datensätze / Schadensentwicklung**

Zur Bewertung eines Schadens ist der zeitliche Verlauf eine wesentliche Eingangsgröße. Es ist von entscheidender Bedeutung, ob sich ein Riss schnell über die Jahre vergrößert oder nahezu stabil in seiner Länge und Ausprägung bleibt. Diese Aussage kann bisher so nicht abgeleitet werden, da keine räumlich exakte Synchronisierung der aufgenommenen Rohdaten erfolgt. Ein Vergleich zwischen zwei Messsystemen bzw. Folgemessungen basiert auf gemittelten 20m bzw. 100m langen Abschnitten. Die Synchronisierung erfolgt auf Basis der GPS-Koordinaten, der gefahrenen Weglänge und des vorhandenen Modells der Verkehrsanlage (Knoten-Kanten-Modell). In Deutschland erfolgt dies nach dem Verfahren

der „standardisierten Netzzuordnung“. Dieses Verfahren ist in einer Software (GeoRohR-aster) implementiert. Die dabei eingesetzten Algorithmen sind allerdings für einen Anwender nicht ersichtlich. Die Synchronisierung erfolgt auf Ebene der gemittelten 100 bzw. 20 m Abschnitte. Dies ist für eine Schadensverfolgung zu grob.

1.4 Zielsetzung

Folgende Methoden sollten eingesetzt werden, um die nachfolgenden Ziele zu erreichen:

Anwendbarkeit auf unterschiedliche Straßenbeläge

Durch Einsatz von Experten für die Analyse von Schadstellen soll ein umfassender Datenbestand aufgebaut werden, der Schäden in möglichst vielen unterschiedlichen Ausprägungen abdeckt. Dabei wird ein iteratives Vorgehen angestrebt. Die Schätzung der Unsicherheit der Schadensdetektoren soll frühzeitig für die Auswahl von besonders gut geeignetem Material für das nachfolgende Labeling eingebunden werden, um das Labeling auf genau jene Straßenstücke zu konzentrieren, die die Qualität des Detektors möglichst stark verbessern, während Schäden, die bereits sicher erkannt werden können, nicht noch zusätzlich benötigt werden und folglich auch nicht gelabelt werden müssen. Dabei soll das an der TU Ilmenau entwickelte Verfahren nach [Seichter 2018] zum Einsatz kommen.

Auf Grundlage dieser Daten werden Convolutional Neural Networks als eine spezielle Form der tiefen Neuronalen Netze als Detektor trainiert. Dabei soll von dem Detektor nicht nur die Lokalisation und Klassifikation von Schadens- und Objektklassen erfolgen, sondern auch die Oberflächenklasse (z.B. Asphalt, Beton, offenporiger Asphalt etc.) geschätzt werden, um oberflächenabhängige Schadstellenklassen behandeln zu können.

Um eine Klassifikation für alle im D-A-CH-Raum gebräuchlichen Schadenskataloge zu ermöglichen, erfolgt die Klassifikation anhand eines, im Rahmen des Projekts auszuarbeitenden, verallgemeinerten Objekt- und Schadenskataloges. Die von dem Detektor auf Grundlage des verallgemeinerten Katalogs erzeugten Schadstellen- und Objektkarten bilden die Grundlage für die sich anschließende länderspezifische Auswertungen.

Modellierung der Unsicherheit der Klassifikation

Bisher konnte sich kein Ansatz zur Modellierung der Unsicherheit in Neuronalen Netzwerken gegenüber alternativen Konzepten als der beste herausstellen, sodass in diesem Vorhaben eine vergleichende Analyse der vielversprechendsten Ansätze notwendig ist, um daraus ein geeignetes Qualitätsmaß für die Sicherheit der Entscheidung abzuleiten. Daher sollen folgende Methoden umgesetzt und ggf. kombiniert werden: Bayessche Inferenz

[Kendall 2017], Ensemblestrategien [Lakshminarayanan 2017], Kalibrierung der Netzwerkausgabe [Guo 2017] sowie eine spezifische Netzwerkausgabecodierung [Kardan 2017].

Zeitliche Analyse von Schadstellen

Grundlage für die zeitliche Analyse von Schäden bildet eine räumliche Synchronisation von zeitlich versetzt aufgenommenen Befahrungssequenzen. Im Rahmen des Projekts soll ein Verfahren für die Herstellung der räumlichen Synchronisation von Oberflächenbildern umgesetzt werden. Aufbauend auf den räumlich synchronisierten Befahrungssequenzen und auf Grundlage der von dem Detektor generierten Schadenskarte, soll die Bewertung der zeitlichen Schadensentwicklung auf Grundlage einer sogenannten „Change Detection Analyse“ erfolgen. Diese Art der Analyse ist ein im Remote-Sensing-Bereich weit verbreitetes Verfahren zur Erkennung von Veränderungen in Bilddaten, die zu verschiedenen Zeitpunkten aufgenommen wurden. Dieses Konzept wird hier adaptiert, um Schadensausprägungen zu verschiedenen Zeitschnitten entlang definierter Kriterien, wie z.B. der betroffenen Fläche, zu vergleichen.

Übertragbarkeit der Lösung auf andere Infrastrukturen

Bei der Übertragung der Lösung auf andere Infrastrukturen soll das bereits vorn erläuterte „Transfer Learning“ zur Wiederverwendung von bereits auf der Domäne Straße gelernten Netzwerkgewichten des tiefen Neuronalen Netzwerkes mit anschließendem Finetuning auf Daten der neuen Domäne eingesetzt werden.

1.5 Vorgehensweise

Die im Projekt ASFaLT angestrebte Automatisierung der Schadstellendetektion soll in allen Ländern des D-A-CH-Raums Anwendung finden können. In der ersten Phase des Projekts stand daher die Analyse und Vereinheitlichung der in den Ländern gültigen Schadensklassen im Fokus (Kapitel 2), um einen länderübergreifenden Schadenkatalog zu definieren und Regeln für die Übertragung in und aus den jeweiligen Ländervorgaben zu erarbeiten. Ausgehend von diesem Schadenkatalog wurden anschließend Labelklassen definiert und ein Labeltool angepasst, um Daten für das Training einer automatisierten Detektion generieren zu können (Kapitel 3). Aufbauend auf diesen Daten konnten anschließend Segmentierungsansätze für die Schadens- und Objektdetektion auf Bildern von Straßenoberflächen umgesetzt werden (Kapitel 4) und Untersuchungen für die Bestimmung von Unsicherheiten auf Bilddaten aus der Straßendomäne durchgeführt werden (Kapitel 5).

Für die Detektion und auch für die länderspezifische Umwandlung wird grundsätzlich zwischen Asphalt- und Betonoberflächen unterschieden. Wie die automatische Detektion der Oberfläche durchgeführt wird, wird daher in Kapitel 6 beschrieben. Die Umwandlung in die länderspezifischen Schadenskataloge und Standards benötigt für manche Klassen noch eine gesonderte Nachverarbeitung, welche anschließend in Kapitel 7 beschrieben wird. Die Integration aller Komponenten in eine Anwendung und die Entwicklung einer Benutzeroberfläche wird in Kapitel 8 genauer beleuchtet. Der mit der Gesamtanwendung mögliche Vergleich zwischen der automatisierten und der klassischen händischen Zustandserfassung schließt sich in Kapitel 9 an.

Die Automatisierung ermöglicht die Schadenserfassung mit einer deutlich höheren räumlichen Auflösung. Die dadurch mögliche Betrachtung der Entwicklung von Schäden auf Bildebene wird in Kapitel 10 genauer betrachtet. Kapitel 11 zeigt zum Abschluss am Beispiel Tunnel auf, wie die Schadensdetektion auch auf andere Domänen übertragen werden kann.

2 Katalogisierung von relevanten Schadensklassen im D-A-CH-Raum

Im Sinne der D-A-CH-Kooperation sollte der zu entwickelnde Schadstellendetektor derart ausgelegt sein, dass er sich möglichst universell in allen drei Ländern anwenden lässt und somit gegebenenfalls Weiterentwicklungen auch in allen Ländern zum Einsatz kommen können. Um diese zu gewährleisten, ist es wichtig, dass der Detektor in der Lage ist alle für die jeweiligen Ländern relevanten Schadensklassen zu detektieren und eindeutig in die Schadenskataloge der Länder zu überführen.

Da die Schadenskataloge der Länder bisher jedoch weitestgehend unabhängig voneinander gepflegt wurden, mussten in einem ersten Schritt zunächst die Regelwerke und Richtlinien der Länder analysiert und gegeneinander abgeglichen werden, um Gemeinsamkeiten aber auch Unterschiede bei der Schadensbeschreibung ermitteln zu können.

Ziel war es dabei klare und möglichst eindeutige Regeln abzuleiten, um Schäden zu beschreiben und gut zwischen unterschiedlichen Schadensarten und -ausprägungen unterscheiden zu können. Dabei wurden nicht nur die Regelwerke der jeweiligen Länder analysiert, sondern auch Erfahrungen aus der Erfassungspraxis eingebracht, um etwaige Diskrepanzen aufdecken zu können. Des Weiteren stand die Abbildung der Schäden auf einen länderübergreifenden Labelkatalog im Zentrum der Überlegung, der die im weiteren Verlauf des Projektes von dem automatisierten System zu detektierenden Klassen festlegt. Wichtig war weiterhin, dass sich die Regelwerke der Länder in den Labelkatalog (3.1) abbilden lassen können und dass auch die Rückprojektion aus dem Labelkatalog auf die Schadenskataloge der Länder möglich ist (Abbildung 1).

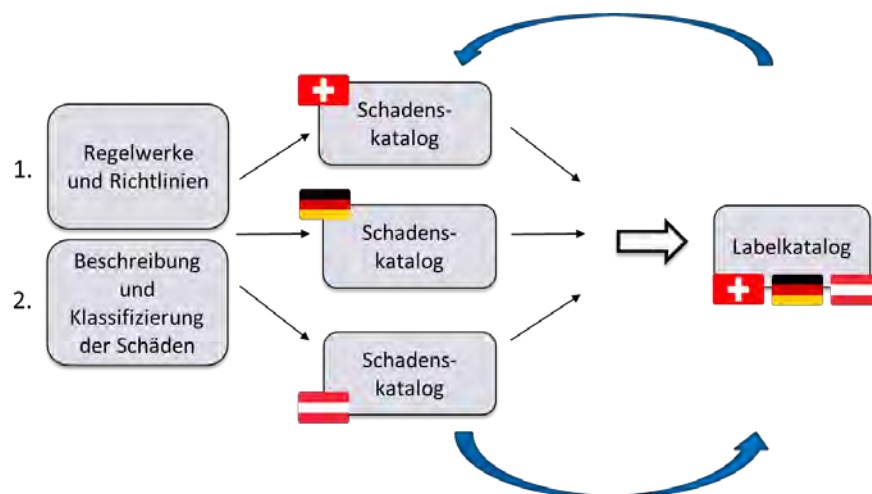


Abbildung 1: Abbildung der länderspezifischen Schadenskataloge auf den länderübergreifenden Labelkatalog. Eine Abbildung muss in beide Richtungen möglich sein.

Die länderspezifischen Standards wurden nach den Oberflächenarten Asphalt und Beton getrennt bearbeitet. Die daraus abgeleiteten Schadenskataloge gehen für jede Schadensart auf die folgenden Aspekte ein:

- bei welchen Ausprägungen des Schadens eine Erfassung erfolgt,
- wie der Schaden erfasst wird,
- welche Größenkriterien gegebenenfalls angewendet werden müssen,
- ob es Sonderfälle gibt und
- wie sich das Schadensbild von anderen Schadensklassen abgrenzt.

Zusätzlich erfolgte außerdem eine Bewertung der Schadensarten nach Wichtigkeit (Abbildung 2). Dabei floss jeweils sowohl die Relevanz eines Merkmals - in Hinblick auf die Bewertung des Straßenzustands - als auch die Häufigkeit des Auftretens des Schadensmerkmals ein. Beide Kriterien wurden zusammen betrachtet, da es bei selten auftretenden Schäden auch deutlich schwerer ist ausreichend Beispielbilder für das Training einer automatisierten Detektion bereitzustellen.

In Abstimmung mit der ASTRA stellt sich heraus, dass der sehr stark differenzierte Schadenskatalog der Schweiz auf nationaler Ebene in der Regel keine Anwendung findet. Vielmehr erfolgt für das nationale Straßennetz nur eine Erfassung der 5 Hauptgruppen. Außerdem wird die stark untergliederte Schadensgruppe der Oberflächenverformungen auf nationaler Ebene durch explizite, nicht bildbasierte Messverfahren erfasst. Da es zusätzlich aktuell in der Schweiz Diskussionen gibt, den sehr umfangreichen Schadenskatalog (22 Klassen) zu vereinfachen, wurde in den ersten Arbeitstreffen des Projekts entschieden, die Schadklassen für die Schweiz eng an den österreichischen Schadenskatalog anzulehnen. Diese Entscheidung beruht auch auf der Tatsache, dass sich die Beschaffung von Bildmaterial von Schweizer Straßen als nicht trivial herausgestellt hat, da die ASTRA nicht direkter Besitzer der Bilddaten ist. Dieses Bildmaterial wäre aber dringend erforderlich, um Schadensklassen, welche nur im Schweizer Standard definiert sind, während des Trainings der Schadstellendetektoren zu berücksichtigen.

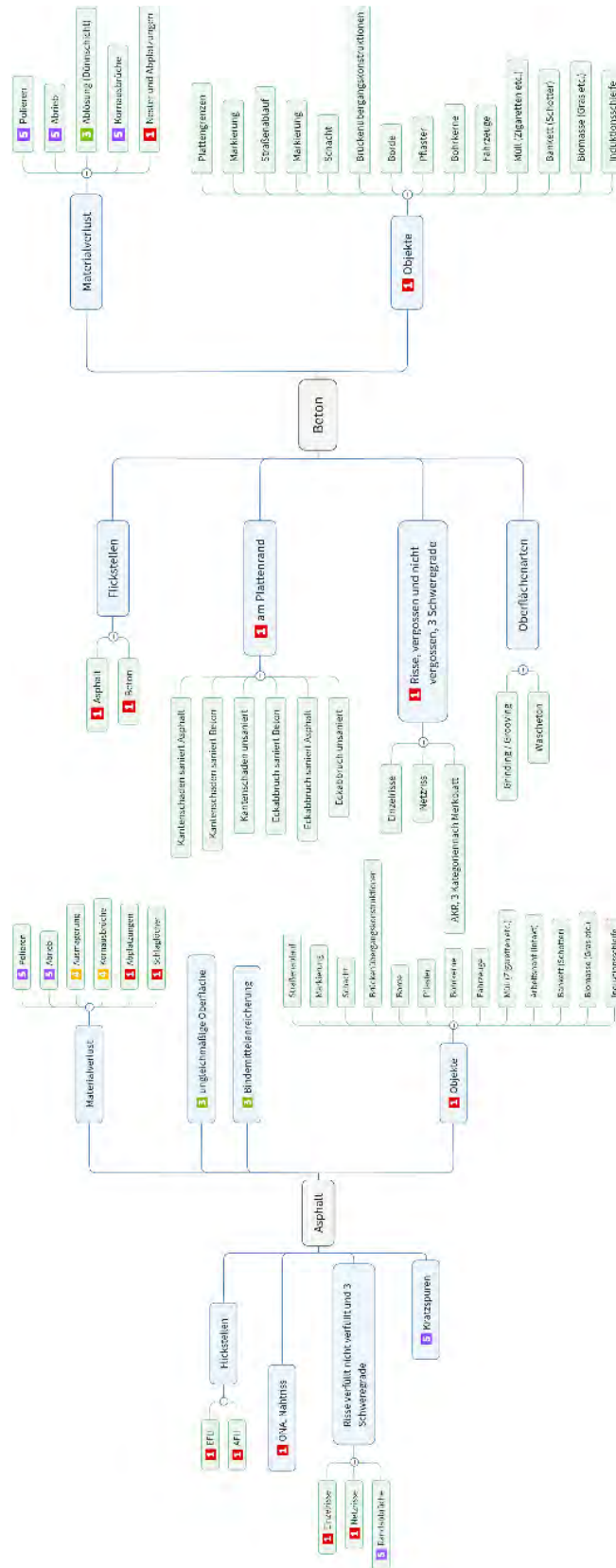


Abbildung 2: Priorisierung der Schadensklassen im D-A-CH-Raum für die Oberflächenarten Asphalt und Beton (1 - hohe Priorität, 5 - geringe Priorität)

Die erarbeiteten Schadenskataloge der Länder stellen jeweils eigene Dokumente dar (als Teil des Arbeitspakets 2) und sind aus Platzgründen nicht Bestandteil dieses Berichts.

3 Datengrundlage für die automatisierte Detektion

Die Realisierung der automatisierten Schadenserkenkung soll im Projekt ASFaLT auf Grundlage von Neuronalen Faltungsnetzwerken erfolgen. Diese wiederum benötigen zunächst Trainingsdaten, um die für die Erkennung erforderlichen Zusammenhänge erlernen zu können. In diesem Kapitel wird daher zunächst betrachtet, wie die Daten für das Training vorbereitet werden müssen (Labeling) und welche Datengrundlage für das Training im Rahmen des Projekts erarbeitet werden konnte.

3.1 Definition von Labelklassen

Auf Grundlage der für alle D-A-CH Länder definierten Schadenskataloge wurde im Projekt ASFaLT der sogenannte Labelkatalog entwickelt. Dieser Katalog enthält die länderunspecifischen, universellen Labelklassen und beschreibt an Beispielen deren Eigenschaften und wie diese für das spätere Training der Detektoren in den Bildern zu markieren sind (Prozess des Labelings - Abbildung 3).

Um die Oberflächenbilder jeweils vollständig beschreiben zu können und um perspektivisch auch das Einbeziehen von weiteren Merkmalen in die Zustandserfassung zu ermöglichen, umfasst der Labelkatalog nicht nur die in den Schadenskatalogen definierten Schadensklassen, sondern auch eine Vielzahl an Objektklassen (z.B. Schächte, Bordsteine oder Markierungen), die regelmäßig auf Oberflächenbildern auftreten.

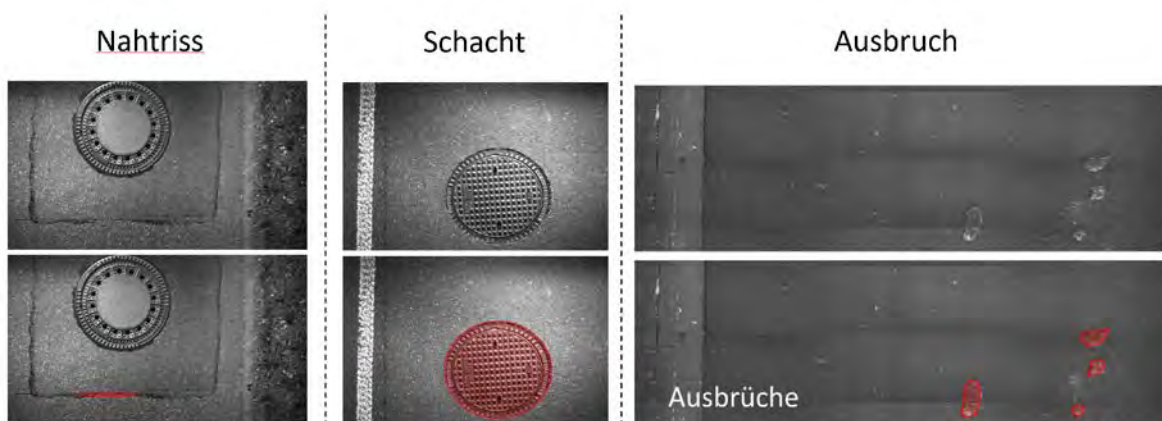


Abbildung 3: Beispielhafte Darstellung von Beispielen für das Labeling der Klassen Nahtriss, Schacht und Ausbruch.

Damit ergibt sich für die Oberflächenart Asphalt eine Gesamtanzahl von 28 Klassen, welche während des Prozesses des Labelings auf den Oberflächenbildern markiert werden müssen (Abbildung 4). Zusätzlich zu den Schadens- und Objektklassen können Bildbereiche auch mit einem VOID Label versehen werden, welches Bereiche im Bild kennzeichnet, die sich nicht einer der Klassen zuordnen lassen, oder bei denen eine Zuordnung nicht sicher getroffen werden kann. Diese Bereiche werden später während des Trainings von dem Detektor nicht berücksichtigt.

| | | | |
|---|-------------------------------|---|----------------------------------|
|  | 0: VOID |  | 15: Markierung |
|  | 1: EFLI |  | 16: Straßenablauf |
|  | 3: AFLI |  | 17: Schacht |
|  | 4: Einzelriss SG 1 (saniert) |  | 18: Brückenübergangskonstruktion |
|  | 5: Einzelriss SG 2 |  | 19: Borde |
|  | 6: Einzelriss SG 3 |  | 20: Pflaster |
|  | 7: Naht |  | 21: Bohrkern |
|  | 8: Abplatzung/Ablösung |  | 22: OBJ Mobil |
|  | 9: Kornausbruch |  | 23: OBJ Stationär |
|  | 10: Ausmagerung |  | 24: Naht intakt |
|  | 11: Kratzspur |  | 25: Bankett |
|  | 12: Ablösung Dünnschicht |  | 26: Biomasse |
|  | 13: ungleichmäßige Oberfläche |  | 27: Induktionsschleife |
|  | 14: Bindemittelanreicherung |  | 28: Normal |

Abbildung 4: Darstellung der im Labelkatalog definierten Schadens- und Objektklassen für die Oberflächenart Asphalt. Schadensklassen sind rosa hinterlegt, während Objektklassen blau hinterlegt dargestellt sind.

3.2 Labeling von Oberflächenbildern

Die im Rahmen der Anforderungsanalyse ermittelten Rahmenbedingungen stellen hohe Anforderungen an das durchzuführenden Labeling. So müssen mit 28 Klassen relativ viele verschiedene Klassen gelabelt werden. Außerdem stellen die zu verarbeitenden Oberflächenbilder mit ihrer Größe von bis zu 60 Megapixeln besondere Anforderungen an eine für das Labeling zu nutzende Software.

Im Rahmen des Auswahlprozesses für eine geeignete Annotierungs-Software wurden nach einer ersten Recherche folgende Tools in Betracht gezogen:

- Coco Annotator [<https://github.com/jsbroks/coco-annotator>]
- RectLabel [<https://rectlabel.com>]
- ScaLabel [<https://scalabel.ai>]
- Labelbox [<https://labelbox.com/>]
- CityScapes Annotation Tool [<https://github.com/mcordts/cityscapesScripts/tree/master/cityscapescripts>]

In einem zweiten Schritt wurden die gegenständlichen Tools von den einzelnen Projektpartnern installiert und auf ihre prinzipielle Eignung hin getestet. Bei lizenzpflichtigen Lösungen

konnte jeweils eine kostenfreie Demovariante herangezogen werden. Die daraus erhaltenen Erkenntnisse bzw. die für das Projekt ASFaLT relevanten Eigenschaften sind in Tabelle 1 ersichtlich.

Tabelle 1: Vergleich der verschiedenen, im Projekt untersuchten Labelling-Anwendungen.

| |  COCO Annotator |  RectLabel |  Scalabel |  Labelbox |  CITYSCAPES DATASET |
|--------------------------|--|---|---|--|--|
| Betriebssystem | Windows / Linux | Mac OS | Windows / Linux | Windows / Linux | Windows / Linux |
| Freeware | ✓ | - | ✓ | - | ✓ |
| Open Source | ✓ | - | ✓ | - | - |
| Vorerfahrungen vorhanden | - | - | ✓ | - | ✓ |
| Browserbasierte Lösung | ✓ | - | ✓ | ✓ | - |
| Lokale Datenablage | ✓ | ✓ | ✓ | - | ✓ |

Das Tool *RectLabel* wurde nicht weiter berücksichtigt, da es nur unter MacOS-Betriebssystemen lauffähig ist und nicht bei allen Projektpartnern Apple-Produkte vorhanden waren bzw. aufgrund der dort bestehenden Hardware-Policy nicht angeschafft werden konnten. Außerdem ist dieses Produkt kostenpflichtig, wobei die geringen Kosten mit 5,50 EUR pro Installation nicht relevant für die abschließende Beurteilung waren.

Das Tool *Labelbox* ist ebenfalls kostenpflichtig, wobei die Preisklasse abhängig von der Anzahl der erstellten Labels ist. Auch dieses Tool wurde nach erfolgter Prüfung nicht weiter in Betracht gezogen, sodass die genauen Kosten für einen breiten Einsatz nicht mehr ermittelt wurden. Gründe für diese Entscheidung waren einerseits der Umstand, dass die zu annotierenden Bilder nicht lokal abgelegt werden können, sondern Cloud-basierend zentral beim Anbieter gespeichert werden, was Fragen zur Datensicherheit aufbringt. Andererseits ist für dieses Tool der Quellcode nicht verfügbar, wodurch es keine Möglichkeit der Optimierung hinsichtlich Bedienung und Konfiguration gibt.

Die Notwendigkeit der individuellen Anpassung des Labeltools auf die speziellen Bedürfnisse im ASFaLT-Projekt stellte sich im Laufe der Recherchen und der Erarbeitung des Labelkataloges (siehe 3.1) als wichtige Grundlage für die Erstellung eines qualitativ hochwertigen Datensatzes gelabelter Bilder heraus, weshalb im laufenden Auswahlprozess in weiterer Folge besonderes Gewicht auf OpenSource-Projekte gelegt wurde.

Aus diesem Grund wurde auch das Tool *Cityscapes* nicht in die engere Wahl genommen, da auch hier der Source Code nicht verfügbar ist.

Die beiden verbleibenden Tools *Coco Annotator* und *ScaLabel* sind beide frei verfügbare OpenSource-Projekte und ermöglichen eine lokale Datenablage. Zusätzlich wird durch die browserbasierte Umsetzung die Flexibilität erhöht und der Installationsaufwand gesenkt. *ScaLabel* war überdies beim Projektpartner AIT bereits bei anderen Projekten im Einsatz, wodurch in diesem Fall Vorerfahrungen vorhanden waren.

Für eine abschließende Beurteilung wurden beide Tools von der TU Ilmenau auf einem zentralen Server aufgesetzt, 10 m-Demobilder eingespielt und die Tools mit jenen Labelklassen konfiguriert, die in diesem Projektstadium bereits definiert waren (z.B. vergossene Risse). Auf diese Art hatten die Projektpartner die Möglichkeit, die Tools unter realen Bedingungen hinsichtlich Bedienbarkeit, Performance, Potential für Erweiterungsmöglichkeiten etc. zu testen.

Das abschließende Feedback fiel zugunsten von *ScaLabel* aus, welches vor allem hinsichtlich der Performance beim Arbeiten mit großen 10 m-Bildern ein besseres Verhalten aufwies.

Nach der Endauswahl wurden die, beim Auswahlprozess gewonnenen Erkenntnisse bezüglich notwendiger Anpassungen von der TU Ilmenau am Tool *ScaLabel* umgesetzt. Dies betraf unter anderem:

- Möglichkeit ins Bild zu zoomen
- Einführen der Labelart „Midline“, die es ermöglicht, Risse zeitsparender zu labeln, als es mit der klassischen Methode ‚Polygon‘ der Fall ist
- Shortcuts für häufig benutzte Befehle einführen
- Erstellen der Pakete sowie einer Installationsanleitung für unterschiedliche Betriebssysteme

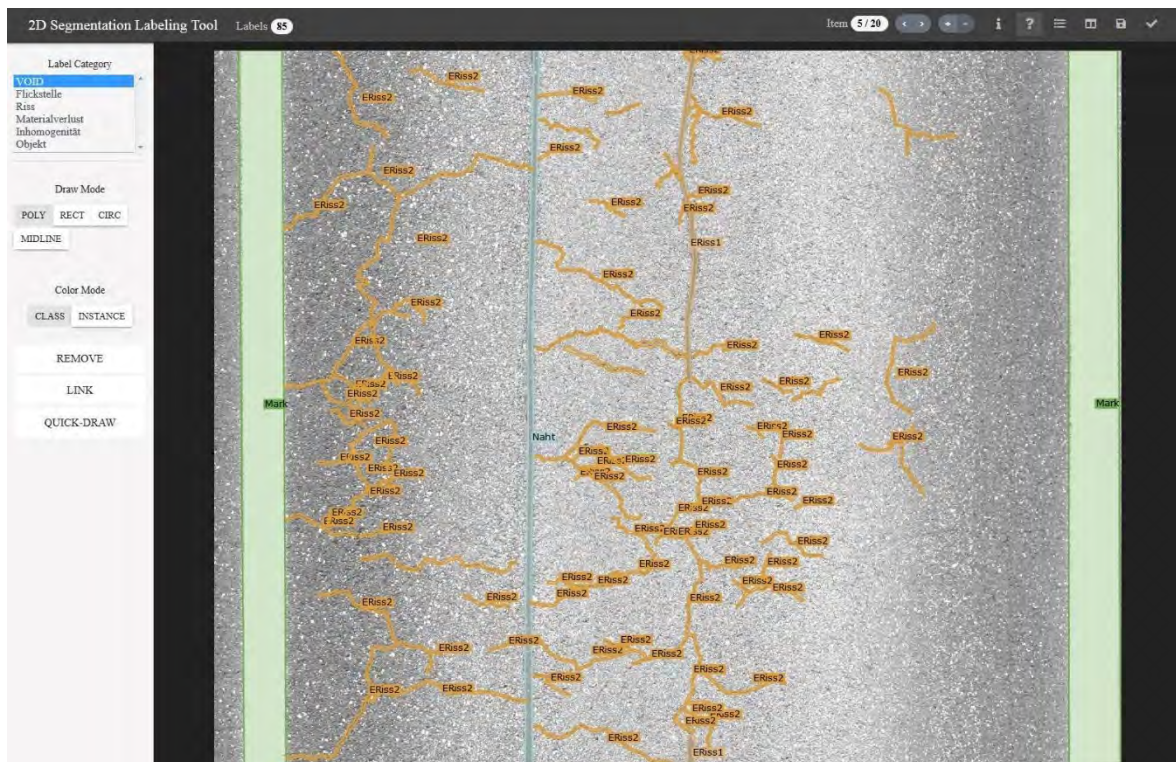


Abbildung 5: Modifizierte Version des Scalabel Tools für das Labeling von Oberflächenbildern. Schäden werden durch Polygone exakt erfasst.

3.3 Datenbasis

Basierend auf dem im Projekt verabschiedeten Labelkatalog und dem im vorigen Kapitel vorgestellten Labeltool erfolgte das Labeling von Schadstellen und Objekten auf Oberflächenbildern von Asphalt- und Betonstraßen. Dabei ist anzumerken, dass sich das im Projekt durchgeführte Labeling deutlich von der, in unterschiedlichen Länderstandards festgelegten, Schadstellenerfassung unterscheidet, da keine exakte räumliche Erfassung der Schäden stattfindet. Um das Training neuronaler Netzwerke zu ermöglichen, werden relevante Schadens- und Objektinstanzen einzeln und „pixelgenau“ erfasst.

Bei der für das Projekt erforderlichen Genauigkeit gestaltet sich das Labeling wesentlich zeitintensiver als bei einem klassischen Erfassungsprozess. Das Labeling eines 10-Meter-Bildes mit vielen Schadstellen kann daher bis zu einer Stunde in Anspruch nehmen.

Die zu labenden Bilder wurden initial von den Projektpartnern AIT und L+P ausgewählt und deckten bereits ein breites Spektrum an Schadensarten ab.

Da die Datensätze während der Projektlaufzeit sukzessive erweitert wurden, ergaben sich verschiedene Bestände an Datensätzen, mit welchen trainiert wurde. Um eine Bewertung der trainierten Detektoren zu ermöglichen, werden die Datensätze jeweils in unterschiedliche Gruppen aufgeteilt. Der Trainingsdatensatz wird ausschließlich für das Training der

Netzwerke verwendet, der Validierungsdatensatz ermöglicht einen Trainingsabbruch und einen ersten Vergleich für den Einsatz unterschiedlicher Parameter. Der Testdatensatz wiederum wird für die finale Bewertung herangezogen.

Die Teildatensätze wurden jeweils derart erzeugt, dass gleiche Straßen jeweils nur in einem der Datensätze vertreten sind und die Datensätze dennoch eine möglichst gleiche Verteilung an Schadens- und Objektklassen aufweisen. Da diese Bedingungen für den weniger umfangreichen Betondatensatz nicht erfüllt werden konnten, ist für den Betondatensatz kein Test-Set verfügbar.

Für die letzte Iteration des Asphalt Datensatzes wurde die Schätzung der Modellunsicherheit genutzt, um weitere Bilder für das Labeling auszuwählen, bei denen das auf den bereits gelabelten Daten trainierte Modell noch eine große Unsicherheit aufweist (Siehe Abschnitt 5.7).

Das Training von Detektoren wurde während des Projekts zunächst auf die Asphaltoberfläche fokussiert, um die Erfahrungen auf das Labeling und die Auswahl an Betondaten übertragen zu können und um der Verteilung der Oberflächenarten in den D-A-CH Straßennetzen Rechnung zu tragen. Die Entwicklung der Datensätze und die in den jeweiligen Teildatensätzen enthaltenen Bildanzahlen sind in Tabelle 2 dargestellt.

Tabelle 2: Entwicklung der Bildanzahl in den für das Training und die Bewertung verwendeten Datensätzen. Die Datensätze tragen als Kürzel jeweils das Jahr und den Monat ihrer Erstellung.

| Datensatz | Training | Validierung | Test |
|------------------|----------|-------------|------|
| 2019_12_Aspphalt | 191 | 8 | 23 |
| 2020_03_Aspphalt | 246 | 24 | 24 |
| 2020_06_Aspphalt | 390 | 39 | 46 |
| 2021_04_Aspphalt | 559 | 147 | 46 |
| 2021_01_Beton | 158 | 41 | - |
| 2021_04_Beton | 307 | 62 | - |

Durch dieses Vorgehen konnten im Rahmen des Projekts 752 Bilder mit über 15.000 Labels für die Oberflächenart Asphalt und 369 Bilder mit über 5.600 Labels für die Oberflächenart Beton gelabelt werden. Die Verteilung der Anzahl an Trainingsinstanzen auf die einzelnen Schadensklassen ist in Abbildung 6 dargestellt.

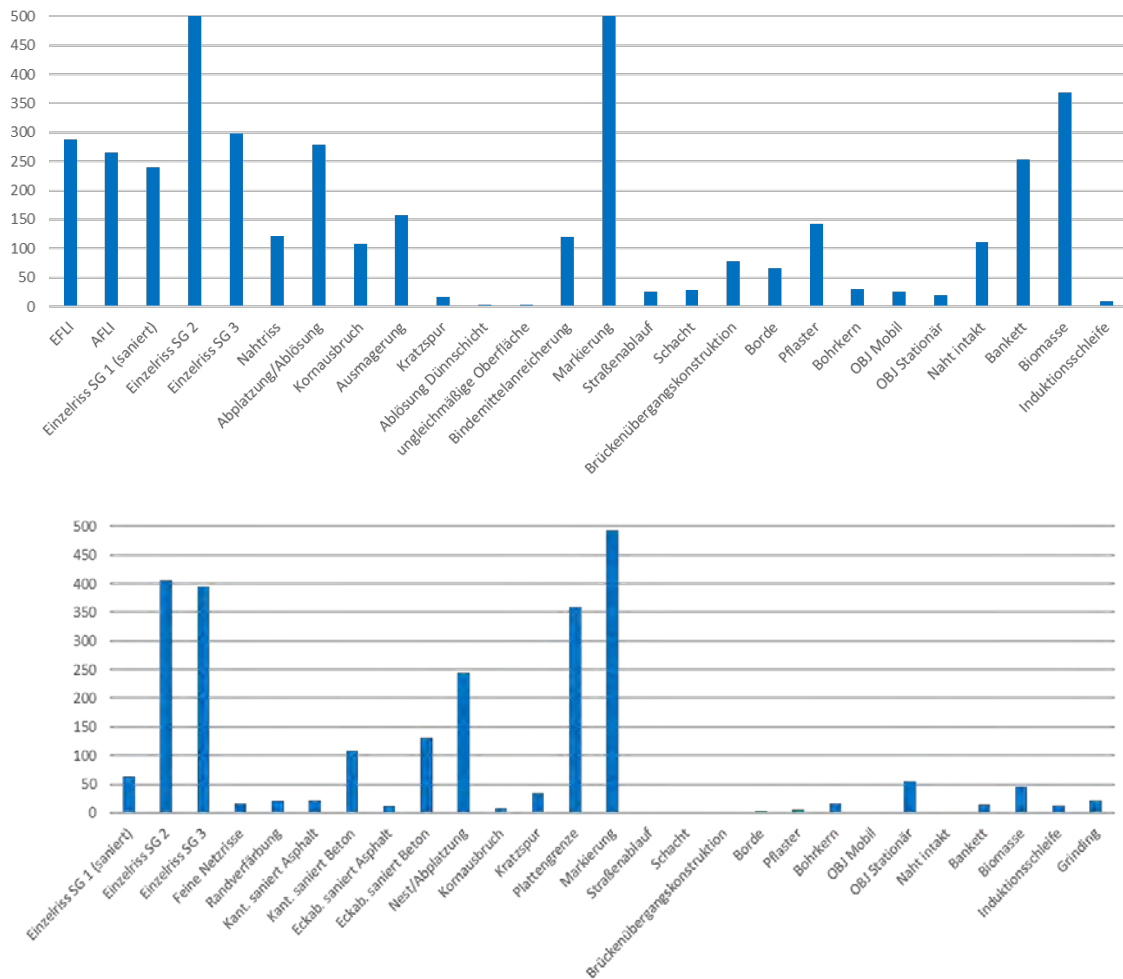


Abbildung 6: Darstellung der gelabelten Anzahl an Trainingsbeispielen pro Klasse (in Prozent zur angestrebten Zielgröße von 200 Instanzen pro Objekttyp). oben - Oberflächenart Asphalt, unten- Oberflächenart Beton

Es ist ersichtlich, dass, insbesondere für die Oberflächenart Beton, nicht für alle Klassen die angestrebte Zielgröße von 200 Trainingsbeispielen erreicht werden konnte.

3.3.1 Adaption von Betonklassen

Bei der Auswertung der ersten Trainingsergebnisse der auf Beton trainierten Detektoren zeigte sich, dass die für das Labeling definierten Klassen nicht ideal sind. Insbesondere in Anbetracht der relativ geringen Anzahl an Trainingsdaten war für die Netzwerke der Zusammenhang zwischen Schäden und ihrer Lokalisation auf der Platte (z.B. Unterscheidung zwischen Eck- und Kantenschäden) nicht erlernbar (Abbildung 7).

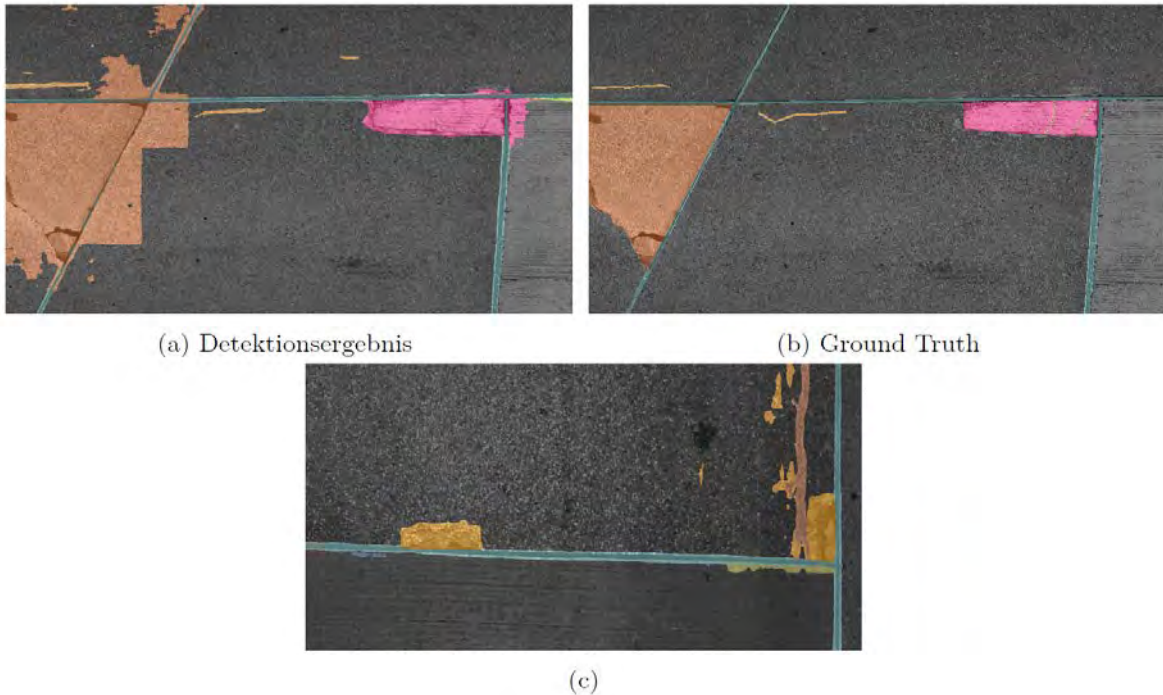


Abbildung 7: Detektionsergebnisse bei Anwendung des Netzwerks auf den Trainingsdaten. Die Klasse Eckabbruch (b)-Dunkelorange begründet sich auf der Detektion des Risses in der Plattenecke. Dieser Zusammenhang scheint für das Netzwerk aus den Trainingsdaten nicht erlernbar zu sein. Die Eckbereiche der angrenzenden Platten werden, aufgrund der identischen Belagstruktur, ebenfalls als Eckabbruch detektiert. Die in (c) dargestellten Kanten- und Eckschäden (beide Orange) haben ein sehr ähnliches Schadensbild. Eine Trennung ist hier für das Netz nicht möglich.

Es wurde daher entschieden, die positionsbezogene Information, die für die reine Ausprägung des Schadens nur eine untergeordnete Rolle spielt, aus den Trainingsdaten zu entfernen. Dazu wurden bei den Labelklassen die in Tabelle 3 aufgelisteten Änderungen vorgenommen.

Tabelle 3: Abbildung der im Labelkatalog definierten Labelklassen auf die für das Training genutzten Klassen

| Ursprüngliche Labelklasse | Abgebildete Labelklassen |
|---|-------------------------------|
| Kantenschaden unsaniert | Riss und/oder Nest/Abplatzung |
| Eckabbruch unsaniert | Riss und/oder Nest/Abplatzung |
| Sanierte Eckabbrüche, Sanierte Kantenabbrüche | Asphalt oder Beton |

Je nach Schadensbild wurden die Klassen damit auf visuell identische Klassen abgebildet, was die Anzahl an Trainingsbeispielen deutlich verbessert und auch die Problemstellung für den Detektor vereinfacht, da gleiche Schadensbilder nicht mehr anhand ihrer Position auf der Platte auf andere Klassen abgebildet werden müssen.

Wie die Schäden für die länderspezifische Bewertung wieder in die jeweils korrekte Schadensklasse konvertiert werden können, wird in 7.3.6 beschrieben.

3.3.2 Klassenmappings

Während der Laufzeit des Projekts wurden für das Training zwischenzeitlich Klassen zusammengefasst oder verschiedene Klassen zu der Klasse VOID gemappt, da die Klassen noch nicht genügend in den Trainingsdaten vorhanden waren.

Die verschiedenen zum Einsatz gekommenen Klassenmappings werden in Tabelle 4 zusammengefasst. Die Klasse VOID ist in allen Klassenmappings vorhanden, es wurde jedoch nie ein Modell trainiert, welches die Klasse VOID präzisieren sollte. Die Klasse VOID legt somit immer nur Bereiche fest, die während des Trainings nicht berücksichtigt werden.

Tabelle 4: Diese Tabelle beschreibt die verschiedenen verwendeten Klassenmappings.

| Name des Klassenmappings | Beschreibung |
|---------------------------------|---|
| Full_Class_Mapping | Alle Klasse sind enthalten. |
| Full_Class_Mapping_2019 | Oberflächenart Asphalt. Alle Klassen abzüglich der Klasse EFLIoF, welche wieder auf EFLI übertragen wird. (EFLIoF steht für „Eingelegte Flickstelle ohne Fugenband“) |
| Classes_For_Training | Oberflächenart Asphalt. Folgende Klassen wurden zu Void gemappt: Ablösung Dünnschicht, Ungleichmäßige Oberfläche, Bohrkern, OBJ Mobil, OBJ Stationär. Ansonsten sind alle weiteren Klassen enthalten. |
| Classes_For_Training_Alt | Oberflächenart Asphalt. Folgende Klassen wurden zu Void gemappt: Kornausbruch, Ausmagerung, Ablösung Dünnschicht, Ungleichmäßige Oberfläche. Die Klassen Einzelriss SG2 und Einzelriss SG3 wurden zusammengefasst. Ansonsten sind alle weiteren Klassen enthalten. |
| 2021_06_Mapping | Oberflächenart Asphalt. Folgende Klassen wurden zu Void gemappt: Ablösung Dünnschicht, Ungleichmäßige Oberfläche. Ansonsten sind alle weiteren Klassen enthalten. |
| 2021_04_concrete_Mapping | Oberflächenart Beton. Zusammenlegung von Klassen nach 3.3.1. Es sind ausschließlich folgende Objektklassen enthalten: Plattengrenze, Markierung, Bohrkern, OBJ Stationär, Induktionsschleife |

4 Automatisierte Schadstellendetektion

Die Anwendung von Segmentierungsverfahren für die Schadstellen- und Objektdetektion im Projekt ASFaLT stellt spezifische Anforderungen an die einzusetzenden Techniken. So sind die zu segmentierenden Oberflächenbilder mit einer Größe von bis zu 60 Megapixeln deutlich größere als bei gängigen bildbasierten Segmentierungsproblemen, welche üblicherweise auf Bildern mit wenigen Megapixeln arbeiten.

Um diesen Umstand Rechnung zu tragen, wurden im Projekt ASFaLT ein zweistufiges Vorgehen angewendet.

In der ersten Phase werden klassische Segmentierungsverfahren verwendet, die jeweils nur auf einem Ausschnitt des Oberflächenbildes arbeiten und somit nur einen begrenzten lokalen Kontext berücksichtigen (4.1).

Anschließend wurde eine eigene Erweiterung der eingesetzten Netzwerkstruktur entwickelt, welche die Integration von erweitertem Kontext ermöglicht (4.3). Ergebnisse der Klassifikation auf den während des Projekts erstellten Daten werden in 4.4 vorgestellt.

4.1 *Eingesetzte Segmentierungsverfahren*

In diesem Abschnitt wird auf die verschiedenen eingesetzten und getesteten Segmentierungsverfahren kurz eingegangen. In Abschnitt 4.1.1 wird das U-Net [Ronneberger, 2015] und in Abschnitt 4.1.2 die Verfahren PSP-Net [Zhao, 2017] und DeepLabV3+ [Chen, 2018] sowie die verschiedenen verwendeten Backbones präsentiert. Abschließend werden in Abschnitt 4.1.3 die untersuchten Fehlerfunktionen vorgestellt.

4.1.1 U-Net

Das U-Net Netzwerk [Ronneberger, 2015], welches auf dem Verfahren aus [Long, 2015] aufbaut, ist eines der ersten Segmentierungsverfahren und wird aber aktuell immer noch erfolgreich vor allem in der Biomedizin eingesetzt. Der generelle Aufbau eines U-Net kann aus Abbildung 8 entnommen werden. Die generelle Idee ist ähnlich zu einem Autoencoder, wobei im ersten Teil des Netzwerks (der Encoder) die Höhe und Breite der Featuremap von Stufe zu Stufe kleiner wird, sich dabei die Tiefe aber immer verdoppelt. In der zweiten Stufe des Netzwerks (der Decoder) nimmt die Höhe und Breite wieder zu und die Tiefe wieder ab, sodass sich am Ende ein Segmentierungsergebnis mit einer Auflösung ähnlich zu der Eingabe ergibt. Durch diese Struktur können auch räumlich auf der Eingabe weit entfernte Informationen zusammen verarbeitet werden, dies ist beispielsweise besonders wichtig für die Erkennung von Flickstellen und intakten Nähten. Weiterhin hat das U-Net einen Shortcut

(grauer Pfeil in Abbildung 8) auf den unterschiedlichen Ebenen des En- und Decoders, um so einem Informationsverlust entgegenzuwirken.

Es gibt viele verschiedene Variationen vom originalem U-Net [Ronneberger, 2015], welches im folgenden Vanilla U-Net genannt wird. Von diesen Variationen wurden im Projekt AS-FaLT das Satellite U-Net¹ und das Xception U-Net² untersucht. Das Satellite U-Net und das Xception U-Net haben beide den Vorteil, dass die Ausgabeauflösung mit der Eingabeauflösung übereinstimmt. Das Satellite U-Net wurde für die Segmentierung von Satelliten- und Drohnenbildern entworfen, wobei dies dem Problem der Straßenbildsegmentierung ähnelt, ebenso hat es den Vorteil, dass die Shortcuts zwischen dem Encoder und Decoder die komplette Featuremap weiterreichen, statt einem Ausschnitt (wie in Abbildung 8). Das Xception U-Net verwendet Blöcke ähnlich dem Xception-Net [Chollet, 2017] auf den einzelnen Ebenen, diese Xception-Blöcke sind weitaus mächtiger als die typische Abfolge von aufeinanderfolgenden Convolutional Schichten wie sie im Vanilla U-Net üblich sind.

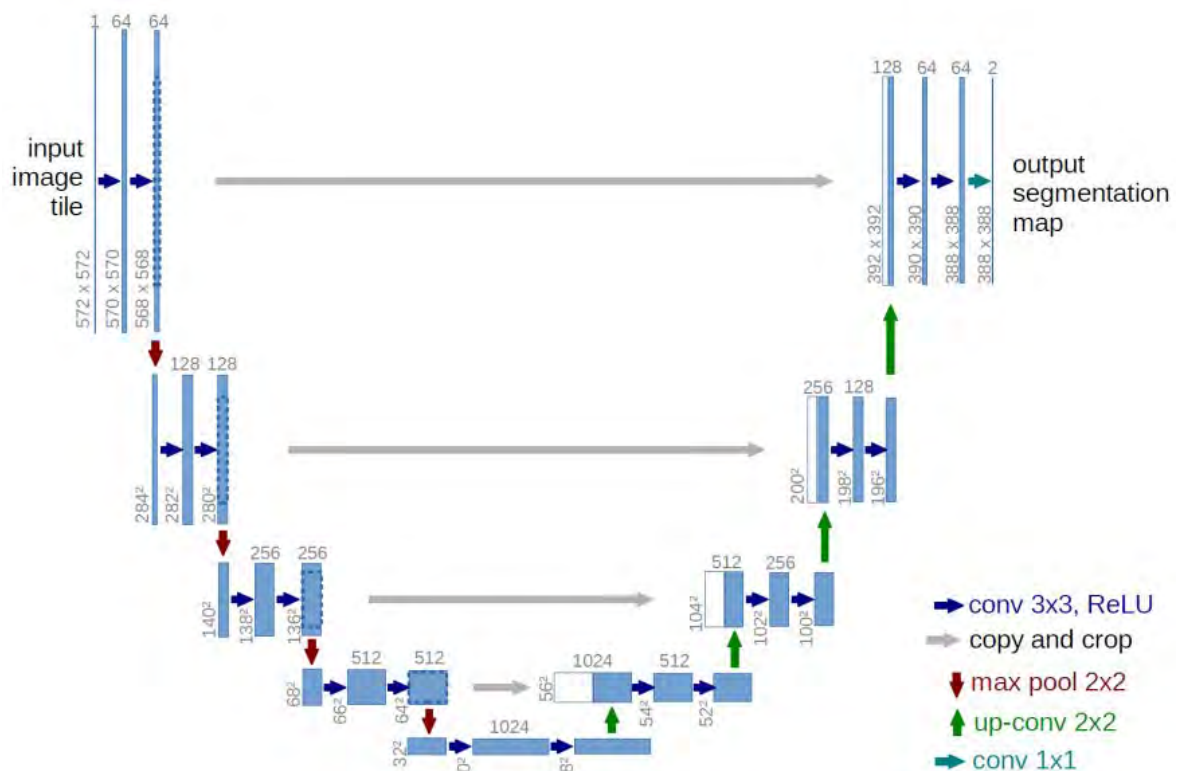


Abbildung 8: Die U-Net Architektur aus [Ronneberger, 2015]. Hier ist das Eingabebild 512×512 Pixel und die Ausgabe 388×388 Pixel groß. Die Auflösung kann bei diesem Architekturaufbau variabel gewählt werden und

¹ <https://deepsense.ai/deep-learning-for-satellite-imagery-via-image-segmentation/>

² https://keras.io/examples/vision/oxford_pets_image_segmentation/

kann daher auch für unterschiedliche Eingabegrößen bei der Staßensegmentierung verwendet werden. Bildquelle: [Ronneberger, 2015]

4.1.2 Typische aktuelle Segmentierungsverfahren

In diesem Abschnitt werden das PSP-Net [Zhao, 2017] und DeepLabV3+ [Chen, 2018] zusammen vorgestellt. Beide besitzen einen ähnlichen Aufbau, welcher derzeit im State of the Art typisch für Segmentierungsverfahren ist. Dieser Aufbau wird im folgenden Abschnitt näher beschrieben. Anschließend werden die verschiedenen untersuchten Backbones, welche als Encoder fungieren, in den nachfolgenden Abschnitten vorgestellt und daraufhin wird auf die Decoder der beiden Segmentierer eingegangen.

Allgemeiner Aufbau

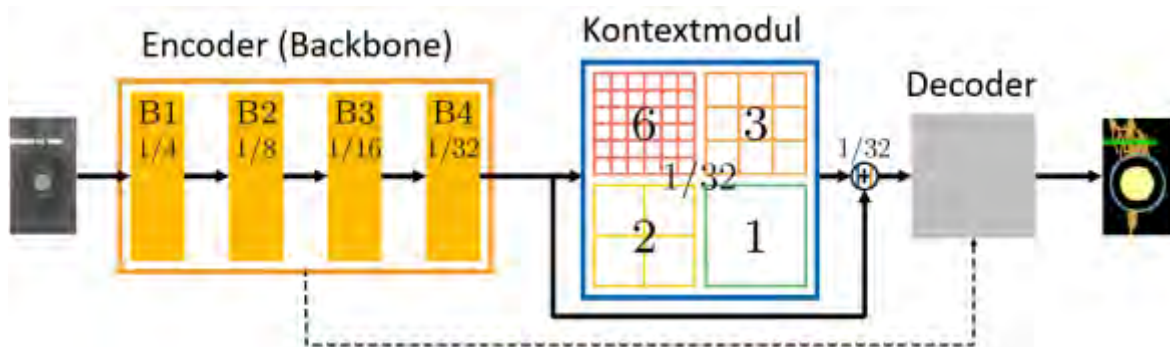


Abbildung 9: Der Aufbau eines typischen Segmentierungsverfahrens. Diese Segmentierungsverfahren setzen sich typischerweise aus einem Encoder, welcher auch in diesem Zusammenhang Backbone genannt wird, einem optionalem Kontextmodul und einem Decoder zusammen.

Ein typischer Aufbau eines Segmentierungsverfahrens kann aus Abbildung 9 entnommen werden. Das Backbone ist in diesem Beispiel ein ResNet-18 [He, 2016] und steht für die typische Reduktion der Eingangsauflösung. In diesem Beispiel ist die Auflösung ein 32tel der Eingangsauflösung am Ende des Encoders. Generell wird als Backbone ein typisches aktuelles Verfahren zur Bildklassifikation verwendet, welches dann an einer geeigneten Stelle abgeschnitten wird. Im optionalen Kontextmodul, welches hier ein Spatial Pyramid Pooling Modul ist, wird die Ausgabe des Encoders auf unterschiedlichen Auflösungsstufen verarbeitet, umso besser Kontextinformationen verarbeiten zu können. Im Decoder werden die Ausgaben des Encoders und des Kontextmoduls verarbeitet und hierbei die Featuremaps wieder vergrößert, bis sie eine Auflösung ähnlich zu der Eingabeauflösung haben. Der Decoder ist in diesem Beispiel grau dargestellt, da sich hier die Segmentierungsverfahren am meisten voneinander unterscheiden.

Verschiedene Backbones

Eingesetzt und untersucht wurden das ResNet [He, 2016], MobileNetV2 [Sandler, 2018] und Xception-Net [Chollet, 2017]. Das MobileNetV2 und das Xception-Net sind dem ResNet sehr ähnlich und verwenden daher beide eine blockartige und mit Shortcuts versehene Netzwerkarchitektur. Aus diesem Grund wird an dieser Stelle nur kurz der Grundgedanke des ResNets präsentiert.

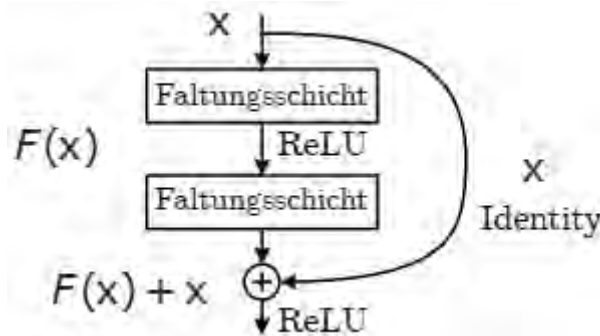


Abbildung 10: Residual-Block des ResNet [He et al., 2016]. Dargestellt ist der schematische Aufbau eines Residual-Blocks, welcher im ResNet verwendet wird. Bildquelle: [He et al., 2016]

Das Problem, welches das ResNet erstmals löste, war das Lernen von tiefen CNNs. Wie die Autoren von [He, 2016] feststellten, half es dem Trainingsgradienten ungemein, wenn sie zusätzlich zu den Faltungsschichten noch Identitätsfunktionen für den Gradientenfluss in das Netzwerk einbauten. So entwickelten sie den Residual-Block, welcher in Abbildung 10 dargestellt ist. Das Netzwerk wurde folglich aus Residual-Blöcken aufgebaut und daher auch ResNet genannt. Mithilfe der Identitätsfunktion (oder auch Shortcut genannt) im Residual-Block müssen nun die einzelnen Blöcke nur noch Veränderungen lernen, wodurch ein einfacheres Training ermöglicht wird und auch deutlich bessere Ergebnisse erzielt werden können.

Das MobileNetV2 verwendet einen abgeänderten ResNet Block und eine gegenüber dem ResNet abgeänderte Anordnung der Blöcke im Netzwerk. Das Ziel hierbei ist es eine schnelle und wenig speicheraufwändige Netzwerkarchitektur zu ermöglichen. Durch den Einsatz vom schnellen MobileNetV2 konnte so im Projekt ASFaLT ein größerer Umfang an Experimenten durchgeführt werden, als wenn nur mit rechentechnisch aufwändigeren, dafür aber besseren, Backbones gearbeitet worden wäre. Nachdem eine gute Parameterwahl auf dem MobileNetV2 gefunden wurde, wurde dann das Training auf dem rechentechnisch sehr aufwändig, dafür aber besserem Xception-Net [Chollet, 2017] wiederholt.

Das Xception-Net ist ähnlich dem ResNet, es verwendet aber Depthwise Separable Convolutions, wodurch es gegenüber dem ResNet auch bessere Ergebnisse erzielt. Das Xception-Net ist das rechen- und speichertechnisch aufwändigste Backbone, das im Projekt AS-FaLT bisher verwendet wurde.

Weiterhin ist noch im Kontext dieses Zwischenberichts anzumerken, dass das verwendete DeepLabV3+ [Chen, 2018] die typischen Backbones zum einen noch mit Atrous Convolution erweitert und zum anderen auch kleinere Veränderungen an den Netzwerkarchitekturen vornimmt. So wird zum Beispiel das Max Pooling im Xception-Net durch weitere Separable Convolutions mit Stride 2 ersetzt. Durch diese Veränderungen wird beim DeepLabV3+ ein einfacheres Training ermöglicht und durch den Einsatz von Atrous Convolution die Verarbeitung von Kontextinformationen verbessert.

Decoder des PSP-Net und des DeepLabV3+

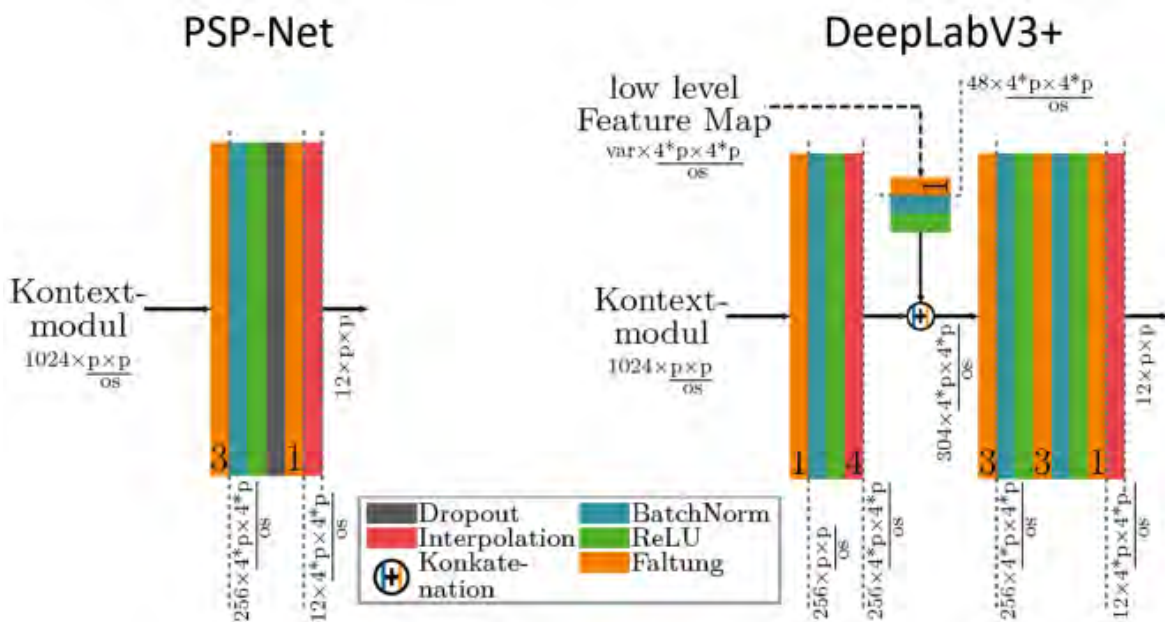


Abbildung 11: In dieser Abbildung ist die schematische Darstellung der Decoder des PSP-Net [Zhao, 2017] und des DeepLabV3+ [Chen., 2018] dargestellt. Der gestrichelte Pfeil beim DeepLabV3+ kommt aus einer früheren Schicht im Encoder und kann mit den Shortcuts in U-Net verglichen werden. In der letzten Schicht wird auf die räumliche Auflösung der Eingabe interpoliert, wobei in dieser schematischen Darstellung die Klassenanzahl auf zwölf festgelegt wurde. Die Eingabegröße (p) und der Output Stride (os) sind hier variabel wählbare Parameter, wobei der Output Stride hier als Zielauflösungsfaktor zu verstehen ist. Das Konzept des Output Stride stammt aus dem DeepLabV3 [Chen., 2017]. Bildquelle: [Engelhardt, 2019]

Abbildung 11 zeigt die Decoder des PSP-Net [Zhao, 2017] und des DeepLabV3+ [Chen, 2018]. Der wichtigste Unterschied zwischen den beiden Decoder ist, dass das DeepLabV3+

in zwei Schritten die Eingaben verarbeitet, bis die räumliche Dimension der Ausgabe mit der räumlichen Dimension der Bildeingabe übereinstimmt.

Nach dem ersten Schritt wird die Featuremap aus einer früheren Schicht im Backbone in den Decoder eingeführt, ähnlich zu den Shortcuts im U-Net. Hierdurch kommt es zu einem geringeren Informationsverlust von kleineren Strukturen, welche sonst bei den kleineren Featuremaps am Ende des Backbons verloren gehen könnten. Dies ist besonders hilfreich für die kleinen Strukturen, wie beispielsweise Rissen bei der Straßensegmentierung.

4.1.3 Verwendete Fehlerfunktionen

Im Projekt ASFaLT wurden neben den verschiedenen Segmentierungsverfahren und Backbones auch verschiedene weitere Fehlerfunktionen getestet. Ein Segmentierungsverfahren wird normalerweise immer mit einer Kreuzentropie-Fehlerfunktion nach einer Softmax-Aktivierung trainiert (Softmax Loss). Hierbei wird bei jedem Pixel berücksichtigt, ob die richtige Klasse prädiziert wurde, richtige Prädiktionen senken den Fehlerwert und falsche Prädiktionen lassen ihn steigen. Durch die starke Unbalanciertheit der Daten (viele Pixel gehören der Normalklasse oder anderen großflächigen Klassen wie Bankett oder Biomasse an) ist es daher nötig, die einzelnen Klassen nach der Häufigkeit, mit der sie im Training vorkommen, zu gewichten (Klassengewichte).

Wie sich aber im Projekt ASFaLT zeigt, könnte ein typisches Lernen mit Softmax Loss und Klassengewichten nicht ausreichen. So neigen die trainierten Modelle oft dazu, viele kleine False Positives für kleine Klassen (wie den Rissklassen) zu prädizieren.

Solche winzigen Risse von wenigen Pixeln Fläche werden aber von keinem Menschen detektiert oder gelabelt. Um dieses Problem zu beheben, wurde die Fehlerfunktion Tversky Loss [Salehi, 2017] in Kombination mit Softmax Loss getestet, welche im nächsten Abschnitt vorgestellt wird.

Weiterhin wurde die Fehlerfunktion Focal Loss [Lin, 2017] in Kombination mit Softmax Loss untersucht. Focal Loss wird näher im folgenden Abschnitt beschrieben.

Tversky Loss

Der Tversky Loss [Salehi et al., 2017] wird mithilfe des Tversky Index berechnet und wird verwendet, um False Positives (FP) oder False Negatives (FN) stärker zu bestrafen, um so die Anzahl der True Positives (TP) zu erhöhen. Der Tversky Index (TI) ist eine Generalisierung des Dice-Koeffizienten und des Jaccard Index und wird wie folgt berechnet:

$$TI = \frac{TP}{TP + \alpha FN + \beta FP}$$

Mit den Parametern α und β können die FN und FP stärker oder schwächer bestraft werden, wobei $\alpha + \beta = 1$ gelten muss. So können mit $\beta > \alpha$ die False Positives stärker bestraft werden, welche auch in ASFaLT ein Problem waren und hiermit behoben werden konnte. Für $\alpha = \beta = 0,5$ ergibt sich der Dice-Koeffizient und für $\alpha = \beta = 1$ der Jaccard Index.

Focal Loss

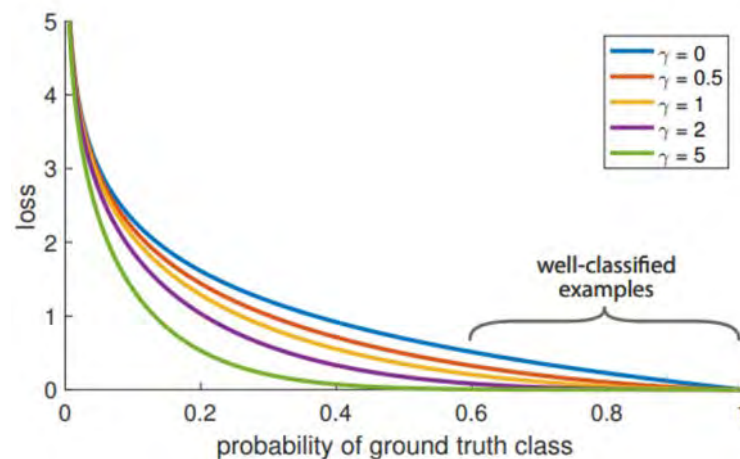


Abbildung 12: Diese Abbildung zeigt verschiedene Focal-Loss-Verläufe für verschiedene Werte von gamma, mit welchem der Focal-Loss-Faktor $(1 - p)^\gamma$ zum Softmax Loss gewichtet wird. Durch Focal Loss kann somit der Fokus auf noch nicht gut gelernte Klassen verstärkt werden. Bildquelle: [Lin, 2017]

Ein typisches Problem beim Training von Segmentierungsverfahren mit stark unbalancierten Daten ist, dass viele Klassen bereits nach wenigen Trainingsepochen schon gut vom Netz beherrscht werden, wodurch der Fehler stark absinkt und weitere Trainingsfortschritte nur noch langsam erzielt werden. Weiterhin ist dies oft der Fall bei den Klassen, welche besonders häufig in den Trainingsdaten vorkommen, wodurch der Fehler noch weiter absinkt. Das Netz beherrscht aber die schwierigen Klassen sowie die Klassen, welche nur selten in den Trainingsdaten vorkommen, an der Stelle im Training noch nicht gut, jedoch ist der Trainingsfehler bereits sehr klein geworden. Dieses Problem könnte auch in ASFaLT vorkommen, da es hier Klassen wie Normal, Bankett, Biomasse, usw. gibt, welche großflächig in den Trainingsdaten vorkommen, leicht zu lernen sind und den Datensatz dominieren.

Dieses Problem soll Focal Loss [Lin, 2017] beheben, indem ein γ -Parameter eingeführt wird, mit dem gesteuert werden kann, wie stark gut prädizierte Klassen bestraft werden sollen. Abbildung 12 zeigt für unterschiedliche γ -Parameter den Focal-Loss-Fehler einer Klasse abhängig davon, wie gut sie prädiziert wird.

Focal Loss (FL) ist wie folgt definiert:

$$FL(p_t) = -(1 - p_t)^\gamma \log(p_t)$$

wobei $p \in [0,1]$, die vom Netz prädizierte Konfidenz für die Ground-Truth-Klasse ist und p_t wie folgt definiert ist:

$$p_t = \begin{cases} p & \text{Klasse stimmt mit Ground Truth überein} \\ 1 - p & \text{sonst} \end{cases}$$

Für $\gamma = 0$ ergibt sich somit der typische Softmax Loss und je größer γ gewählt wird, desto mehr Fokus wird auf die falsch klassifizierte Prädiktionen gelegt.

4.2 **Bewertungsmaße**

In diesem Abschnitt werden die verschiedenen in diesem Projekt verwendeten Bewertungsmaße vorgestellt. In Abschnitt 4.2.1 wird die Mean Intersection over Union (mIoU), welche das typische Bewertungsmaß in der Segmentierung ist, kurz beschrieben. Weiterhin wird in Abschnitt 4.2.2 der Occurrence Score vorgestellt. Dieser Occurrence Score wurde eigens für die Bewertung der Straßenuntergrundsegmentierung entwickelt und soll den Bewertungen nach ZEB-Richtlinien ähneln.

4.2.1 **Mean Intersection over Union**

Die Mean Intersection over Union (mIoU) ergibt sich aus dem Durchschnitt der Intersection over Union (IoU) aller Klassen. Hierbei ist die IoU einer Klasse wie folgt über die True Positives (TP), False Positives (FP) und False Negatives (FN) definiert:

$$IoU = \frac{TP}{TP + FP + FN}$$

Die Prädiktionen werden in einer Konfusionsmatrix akkumuliert, aus welcher dann die mIoU berechnet werden kann. Somit kann die mIoU Werte zwischen null und eins annehmen, wobei je größer der Wert ist, desto besser ist die Prädiktion.

4.2.2 **Occurrence Score**

Die IoU stellt ein probates Mittel für die Beurteilung der Segmentierungsqualität dar und findet daher in der Literatur sehr verbreitete Anwendung.

Die IoU ist jedoch nur begrenzt sinnvoll, um die Qualität der Detektion im Kontext des Projekts ASFaLT zu beurteilen, da ihr Fokus auf der Bewertung der flächigen Übereinstimmung zwischen Ground-Truth und der tatsächlichen Detektion liegt. Sollte beispielsweise eine bestimmte Klasse von einem Detektor immer mit einem Rand von wenigen Pixeln detektiert

werden, äußert sich das direkt in einer sinkenden mIoU. Da jedoch beispielsweise Risse sehr schmal sind und auch das Labeling nicht exakt bis zum letzten Pixel eines Risses erfolgen kann, sind, selbst bei einem optisch sehr guten Ergebnis, auf der Klasse Riss kaum Werte größer als 0.3 für die IoU zu erreichen.

Im Kontext der Anwendung ist es viel entscheidender, dass Detektoren synchron mit der Ground-Truth in gleichen Regionen anschlagen und gleiche Regionen als Normal klassifizieren. Um diesen Umstand Rechnung zu tragen, wurden im Projekt ASFaLT der Occurrence Score entwickelt. Bei diesem Score wird das Detektionsbild mit einem Raster fester Unterteilung, beispielsweise von einem Meter, überlagert (Abbildung 13).

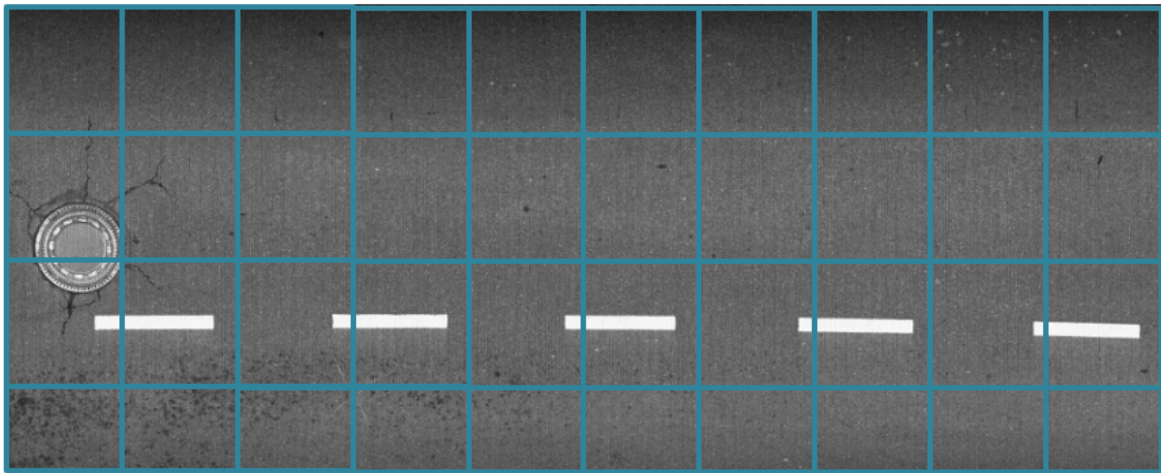


Abbildung 13: Diese Abbildung zeigt die Einteilung eines kompletten Bilds in 1-Meter Kacheln, welcher für die Bewertung mit dem Occurrence Score verwendet werden.

Für die sich ergebenden Rasterzellen wird überprüft, ob Detektion der Klasse c aus n Klassen jeweils in der Ground-Truth und in der Detektion vorhanden sind. Ist dies der Fall, wird, unabhängig von der Häufung, eine positive Übereinstimmung gewertet. Wenn dahingegen eine Klasse nur in der Ground-Truth oder in der Detektion vorhanden ist, wird dies als Fehler gewertet. Für die Berechnung des Occurrence Score occ werden für jede Rasterzelle i ein Vektor gt_{occ} über die Ground-Truth und ein Vektor d_{occ} über die Detektion gebildet. Beide Vektoren besitzen die Dimension n und kodieren in jeder Dimension, ob die Klasse c in $[1, n]$ in der jeweiligen Rasterzelle vorhanden ist:

$$gt_{occ} = (i_1, \dots, i_n), \text{ mit } i_j = 1, \text{ wenn } c \text{ in Ground - Truth - Zelle } i, \text{ sonst } 0$$

$$d_{occ} = (i_1, \dots, i_n), \text{ mit } i_j = 1, \text{ wenn } c \text{ in Detektions - Zelle } i, \text{ sonst } 0$$

Basierend auf diesen beiden Vektoren kann anschließend der Occurrence Score occ_i für Rasterzelle i berechnet werden:

$$occ_i = \frac{\sum_{j=1}^n eq(gt_{occ}(j), d_{occ}(j))}{n}, \text{ mit } eq(a, b) = 1, \text{ wenn } a = b, 0 \text{ sonst}$$

Bei der Berechnung des Occurrence Score für ein ganzes Bild wird das gewichtete Mittel aller Occurrence Scores der Gridzellen entsprechend ihrer Fläche bestimmt. Der so berechnete Score wird im Folgenden als Micro Occurrence Score bezeichnet. Alternativ lässt sich auch der Macro Occurrence Score bestimmen. Bei diesem Score wird der Occurrence Wert für jede Klasse einzeln bestimmt (für das gesamte Bild) und anschließend über die Scores aller vertretenen Klassen gemittelt. Der Micro Score entspricht dabei am ehesten dem visuellen Eindruck. Klassen, die im Bild oft vorkommen, haben auch einen hohen Einfluss auf dem Micro Score des Bildes. Der Macro Score fällt hingegen oft schlechter aus, weil alle Klassen, die im Bild vorhanden sind oder detektiert wurden, gleich gewichtet sind. Wird also z.B. im Bild eine Instanz einer seltenen auftretenden Klasse nicht detektiert, ist der Score für diese Klasse 0. Dieser Score fließt mit gleichem Gewicht in den Macro Score ein, wie eine Klasse, die mit 100 Instanzen auf dem Bild vertreten ist. Entsprechend hat die selten auftretende Klasse, die oft auch schlechter detektiert wird, einen großen Einfluss auf den Gesamtwert.

4.3 Anpassung zur Berücksichtigung von erweitertem Kontext

Einige der im Labelkatalog definierten Klassen lassen sich ohne größeren lokalen Kontext nur schwer eindeutig zuordnen. So ist bei einigen Klassen beispielsweise auch die Position innerhalb des Straßenraums entscheidend, um diese sicher zuordnen zu können.

Ein weiteres Problem ergibt sich in der Straßendomäne für die Oberflächenbilder, welche nicht durch Linienkameras aufgenommen wurden. Diese Oberflächenbilder werden aus Einzelbildern zusammengesetzt, so dass an den Übergängen zwischen den Einzelbildern, je nach Beleuchtungssituation, sehr starke Kontrastkanten entstehen können.

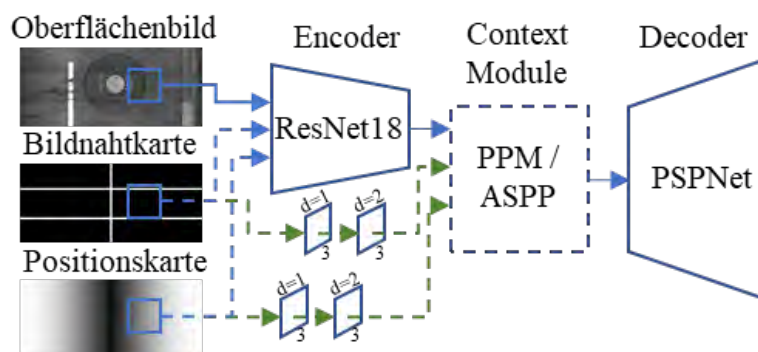


Abbildung 14: Überblick über die Netzwerkarchitektur für die Kodierung von zusätzlichem Kontext. Die Bildnahtkarte und die Positionskarte können optional als zusätzlicher Inputkanal (blau gestrichelt) oder direkt als Input für das Context Module (grün gestrichelt) verwendet werden.

4.3.1 Positionskarte

In [Tilborghs 2020] wurde bereits für eine andere Anwendungsdomäne gezeigt, dass lokale Informationen die Segmentierungsleistung der Faltungsnetzwerke verbessern können.

In Anlehnung daran wurde auch im Projekt ASFaLT die Verwendung von Positionsinformation in Form von Positionskarten untersucht, welche die horizontale Position des aktuellen Bildausschnitts im Oberflächenbild kodieren. Dazu nimmt die Positionskarte im Zentrum des Oberflächenbildes den Wert 0 an und steigt zu den Rändern hin linear auf den Wert 1 an (Abbildung 14).

Die Positionskarte kann dabei auf zwei unterschiedliche Arten in das Netzwerkmodell integriert werden. Zuerst kann die Karte als zusätzlicher Inputkanal verwendet werden, sodass das Netzwerk erlernen kann den Input zu nutzen, falls dieser von Vorteil ist. Um das Einbeziehen von Informationen aus dieser Karte zu forcieren, kann die Karte jedoch auch direkt in das Kontextmodell eingespeist werden. Um den Kontext innerhalb der Karte zu erhöhen, wird die Karte dazu zuerst durch zwei Faltungs-ReLU-Block geleitet (mit Filtergröße 3 und den Dilationsraten 1 (erster Block) und 2 (zweiter Block)).

4.3.2 Bildnahtkarte

Um auch die Kontrastkanten, welche an Bildübergängen zwischen Einzelbildern entstehen, explizit kodieren zu können, müssen diese zuerst detektiert werden. Dazu wird das in Abbildung 15 visualisierte Vorgehen angewendet. Zuerst wird auf einer geglätteten Version des Oberflächenbildes ein in vertikale Richtung orientierter Sobel-Filter angewendet, um horizontale Kanten zu finden. Das entstandene Kantenbild wird in Spaltenrichtung summiert und durch eine diskrete Fast-Fourier-Transformation (FFT) in den Frequenzraum transformiert. Wenn sich in dem für Stitchingkanten typischen Frequenzbereich (10-14 Bilder -> 9-13 Stitchingkanten) ein Maximum finden lässt, wird die Projektion zurück in den Bildraum durchgeführt und das Ergebnis durch eine lokale Maximumssuche noch feinjustiert.

Da die horizontalen Kontrastkanten immer an festen Positionen zu finden sind (in der Mitte bei zwei Kameras, bei 1/3, 2/3 bei 3 Kameras), ist in horizontaler Richtung keine Suche nötig und die Kanten können ja nach Kamerasystem fest eingetragen werden.

Die sich ergebende Bildnahtkarte kann, analog zu 4.3.1 als zusätzlicher Inputkanal oder direkt in das Kontextmodul integriert werden.

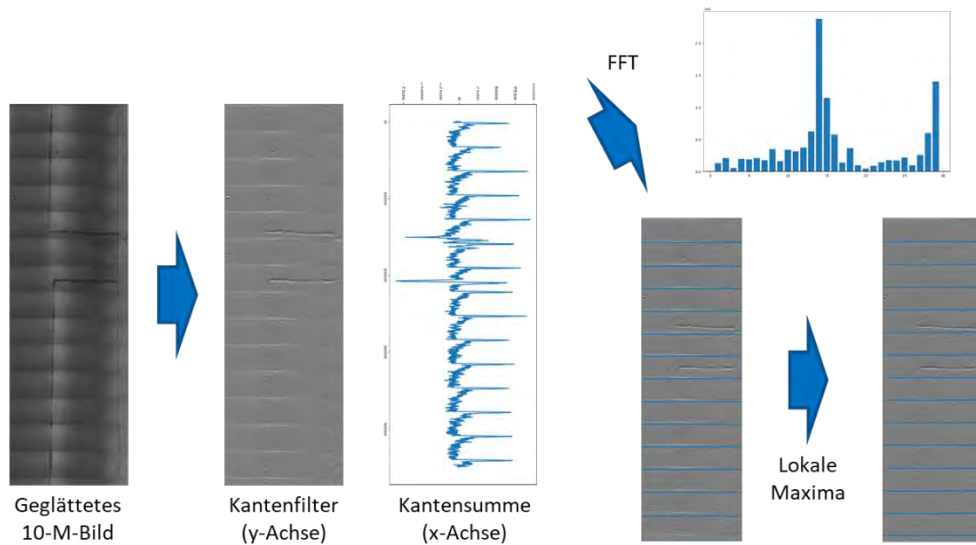


Abbildung 15: Detektion von Bildnähten - Zuerst wird ein vertikal orientierter Kantenfilter auf das Bild angewendet. Auf dem Kantenbild werden anschließend die Zeilensummen berechnet und das entstehende 1-dimensionale Signal mit einer FFT in den Frequenzraum transformiert, um periodische Signale zu finden. Durch Rückprojektion des Maximums in den Bildraum und eine anschließende lokale Maximumssuche werden die Stitchingkanten gefunden.

4.3.3 Ergebnisse

Bei experimentellen Untersuchungen (Tabelle 5) zeigte sich, dass die Positionskarte keine signifikanten Verbesserungen im Vergleich zu einem Netzwerk, welches ohne Anpassungen trainiert wurde (Baseline), erzielen konnte.

Tabelle 5: Erzielte Ergebnisse der Netzwerke mit Kontext in Form von Positionskarten oder Bildnahtkarten. Insbesondere die Integration von Bildnahtkarten hat einen positiven Einfluss auf die Detektion von Schäden und Objekten.

| Methoden | Occurrence Score | mIoU |
|--------------------------------|------------------|-------|
| Baseline | 0,632 | 0,586 |
| Positionskarte (Input) | 0,614 | 0,598 |
| Positionskarte (Context) | 0,637 | 0,577 |
| Bildnahtkarte (Input) | 0,640 | 0,589 |
| Bildnahtkarte (Context) | 0,662 | 0,585 |

Ein Grund dafür liegt darin, dass Klassen, die von der Position innerhalb des Bildes besonders profitieren (z.B. Bordstein), bereits sehr gut erkannt werden. Die Integration der Bildnahtkarten hingegen verbessert die Detektion an den Bildnähten deutlich und führt folglich zu einer deutlichen Steigerung des Occurrence Scores. Diese Steigerung ist insbesondere

bei direkter Integration der Karte in das Kontext-Modul zu erkennen, da auf diese Weise die Information direkt in tiefere Verarbeitungsschichten einfließt und das Lernen mit wenigen Daten erleichtert.

Weitere Details zu dem umgesetzten Verfahren und zu den Ergebnissen, wurden im Rahmen des Projekts unter [Stricker 2021] publiziert.

4.4 Segmentierungsergebnisse

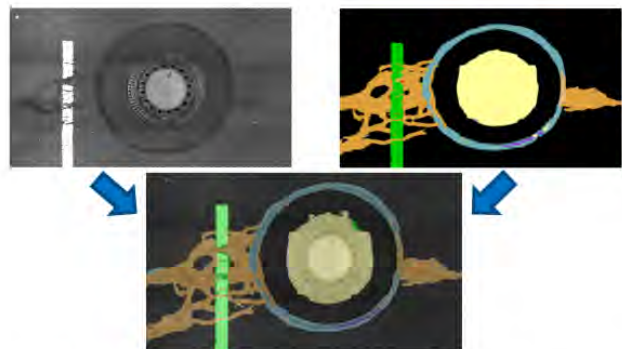
Im Folgenden wird im Abschnitt auf die Visualisierung der Ergebnisse eingegangen. Anschließend wird eine Zusammenfassung der verschiedenen Experimentier- und Segmentierungsergebnisse geliefert. Dabei wird zunächst in Abschnitt 4.4.2 auf die Voruntersuchungen auf den alten Datensätzen und in Abschnitt 4.4.3 auf die Voruntersuchungen auf dem aktuellen Datensatz eingegangen. Anschließend wird in Abschnitt 4.4.4 die Modell- und Parameteroptimierung auf dem aktuellen Datensatz zusammengefasst. Die Ergebnisse des besten trainierten Modells, welches bei diesen Optimierungen gefunden wurde, werden im letzten Abschnitt 4.4.5 präsentiert.

4.4.1 Vorgehensweise bei der Visualisierung der Ergebnisse auf Asphalt Daten

Abbildung 16 zeigt die verschiedenen Klassenlabels, welche im Projekt ASFaLT festgelegt wurden und welche bei der Segmentierung gelernt werden sollten. Mit den zu den Klassen zugehörigen Farben in Abbildung 16 werden die Labels und die Netzwerkausgaben für Asphalt Daten in diesem Ergebnisbericht visualisiert. Die Abbildung 16 zeigt weiterhin ein Beispiel für die Darstellung der Netzwerkausgabe in Kombination mit den Ground-Truth-Daten, diese Darstellungsform wird ebenfalls in diesem Ergebnisbericht verwendet.



(a) Klassenlabels



(b) Beispiel für die Darstellung einer Netzwerkausgabe

Abbildung 16: Teil (a) zeigt die verschiedenen gelabelten Klassen in den Trainingsdaten neben den Farben, welche für die Darstellung der Ergebnisse und Netzwerkausgaben verwendet wurden. Teil (b) zeigt ein Beispiel für die Darstellung einer Netzwerkausgabe und die einzelnen Bestandteile. Hierbei wurde die Netzwerkausgabe mit einem Alpha von 0,5 auf das Eingabebild, welches ebenfalls mit einem Alpha von 0,5 verändert wurde, gelegt. Die Darstellung der Labels erfolgt analog.

4.4.2 Voruntersuchungen auf alten Datensätzen

Zu den alten Datensätzen gehören alle untersuchte Datensätze, welche in Abschnitt 3.3 aufgelistet sind, bis auf den neusten Datensatz und den Datensatz „2020_06 AIT & L+P“ (2020_06). Auf diesen alten Datensätzen wurden Untersuchungen vorgenommen, bei welchen umfangreichen Parameteroptimierungen nicht im Fokus standen, sondern nur Experimente, um generelle Trends zu erkennen und passable Modelle für die Unsicherheits-schätzung zu trainieren. Dies wurde getan, um zu erkennen, für welche Klassen noch nachgelabelte werden sollte und wie der Datensatz im besten Fall erweitert werden könnte. Bei diesen Untersuchungen wurden neben den verschiedenen Datensätzen noch folgende Parameter festgehalten oder variiert.

- Segmentierungsverfahren: Hierbei war der Backbone immer ein ResNet-18 und das Kontextmodul immer ein Spatial Pyramid Pooling Modul. Der Decoder war entweder der Decoder des PSP-Net Moduls oder des DeepLabV3+.
- Klassenmappings: Die verschiedenen Klassenmappings, welche in Abschnitt 3.3.2 beschrieben sind, wurden verwendet.
- Backbone-Initialisierungen: Die verschiedenen Initialisierungen waren entweder zufällige Gewichtsinitialisierung oder eines der vielen verschiedenen auf Patches vor-trainierten Klassifizierungen.
- Trainingsparameter: Die üblichen Parameter wie Lernrate, Lernraten-Scheduler, Batch Size, Optimierungsalgorithmus und Eingabegröße wurden variiert.

In der folgenden Tabelle sind die besten Ergebnisse aus den Voruntersuchungen für die verschiedenen Datensätze und Klassenmappings für die alten Datensätze aufgelistet:

Tabelle 6: In dieser Tabelle sind die besten mIoU Validierungsergebnisse für die verschiedenen Voruntersuchungen auf den verwendeten Datensätzen und Klassenmappings aufgelistet.

| Datensatz | Klassenmapping | mIoU |
|-------------------|--------------------------|-------|
| 2019_12 AIT & L+P | Classes_For_Training | 0,377 |
| | Classes_For_Training_Alt | 0,464 |
| 2019_12 L+P | Classes_For_Training | 0,389 |
| | Classes_For_Training_Alt | 0,457 |
| 2019_12 AIT | Classes_For_Training | 0,444 |
| | Classes_For_Training_Alt | 0,467 |
| 2020_03 AIT & L+P | Classes_For_Training_Alt | 0,482 |
| 2020_03 L+P | Classes_For_Training_Alt | 0,443 |

4.4.3 Voruntersuchungen auf dem Datensatz „2020_06 AIT & L+P“

In den Voruntersuchungen auf dem Datensatz „2020_06 AIT & L+P“ wurde eine umfangreiche Parametersuche für die Augmentierung vorgenommen. Hierbei wurde weiterhin das Modell unter anderem auch mit Bildern trainiert, welche auf 450 Pixel pro Meter skaliert wurden. Dies wurde aus zweierlei Gründen getan. Zum einen hat der Großteil der Bilder auf dem IT-ZEB Server³ 450 Pixel pro Meter, dies ist für spätere Untersuchungen und den menschlichen Vergleich wichtig. Zum anderen ist somit das Training ein Vielfaches schneller, da die Trainingsbilder mindestens 900 Pixel pro Meter haben.

³ Siehe für weitere Informationen: <http://itzeb.heller-ig.de/>

Hierdurch konnte eine viel größere Anzahl an Parametern getestet werden, wodurch sich auch bessere Segmentierer ergeben haben. Die besten Parameter für typische Augmentierungen sind in der folgenden Tabelle aufgelistet (falls sich keine beste Parametereinstellung ergeben hat, wird eine Auswahl an guten Ergebnissen aufgelistet):

Tabelle 7: In dieser Tabelle werden die besten gefundenen Parameter für die verschiedenen untersuchten Wertebereiche der Augmentierungsparameter aufgelistet. Label der Flickstelle steht hierbei dafür, ob das Trainingslabel nur über den Flickstellenrand geht oder flächig die komplette Flickstelle ausfüllt. Die Parameter Helligkeit, Kontrast und Rauschen wurden verändert, wobei die Veränderung zwischen null und eins erfolgen kann. Die Rotation ist in Grad angegeben, wobei die Rotation zusätzlich zu einem zufälligen horizontalen oder vertikalen Flippen erfolgt. Die Eingabe ist immer quadratisch und somit beschreibt die Eingabegröße hier die Länge einer Quadratseite in Pixel.

| Parameter | Getesteter Wertebereich | Beste Parametereinstellung |
|--------------------------------|---|----------------------------|
| Batchsize | [5, 8, 16] | [5, 8, 16] |
| Label der Flickstelle | [Rand, Flächig] | [Rand] |
| Helligkeit, Kontrast, Rauschen | [[0,0, 0,0, 0,0], [0,1, 0,3, 0,0], [0,2, 0,5, 0,02], [0,0, 0,1, 0,3], [0,02, 0,2, 0,5]] | [0,2, 0,5, 0,02] |
| Rotation | [0, 20, 30, 40] | [0] |
| Eingabegröße | [480, 960] | [480, 960] |

4.4.4 Modell- und Parameteroptimierungen auf dem Datensatz „2020_06 AIT & L+P“

In diesem Abschnitt werden die Modell- und Parameteroptimierungen auf dem Datensatz „2020_06 AIT & L+P“ zusammengefasst. Wie im vorherigen Abschnitt wurden die Modelle größtenteils mit Bildern trainiert, welche auf 450 Pixel pro Meter skaliert wurden. Die untersuchten Parameter werden im Folgenden aufgelistet:

- Segmentierungsverfahren: Als Segmentierungsverfahren wurden alle Verfahren, welche in Abschnitt 4.1 beschrieben sind, getestet. Dazu gehören die verschiedenen U-Net-Verfahren, das PSP-Net, das DeepLabV3+ und die verschiedenen Backbones (Res-Net, MobileNetV2, XceptionNet)
- Klassenmappings: Als Klassenmapping wurde das „Full_Class_Mapping“, welches in Abbildung 16 beschrieben ist, angewendet.
- Fehlerfunktionen: Als Fehlerfunktion wurde, neben dem typischen Softmax Loss, die Fehlerfunktionen Focal Loss und Tversky Loss untersucht.
- Trainingsparameter: Als Trainingsparameter wurden zum einen für die Augmentierung die besten Parametereinstellungen, welche in Abschnitt 4.4.2 zusammengefasst sind, verwendet. Zum anderen wurden die weiteren üblichen Parameter wie Lernrate, Lernraten-Scheduler, Epochenanzahl und Optimierungsalgorithmus variiert.

In der nachfolgenden Tabelle werden die besten Ergebnisse für die verschiedenen Segmentierer und weiteren Parameter aufgelistet.

Tabelle 8: In dieser Tabelle sind die besten mIoU Validierungsergebnisse für die verschiedenen Untersuchungen auf dem Datensatz „2020_06 AIT & L+P“ mit „Full_Class_Mapping“ aufgelistet.

| Untersuchung | Beschreibung der Parameter | beste mIoU |
|---|--------------------------------|--------------|
| Segmentierer (Skalierung=450) | Vanilla U-Net | 0,264 |
| | Satellite U-Net | 0,416 |
| | Xception U-Net | 0,420 |
| | PSP-Net mit ResNet-18 | 0,449 |
| | DeepLabV3+ mit MobileNetV2 | 0,477 |
| | DeepLabV3+ mit Xception-Net | 0,491 |
| Fehlerfunktion (Skalierung=450) | Softmax Loss (SL) | 0,491 |
| | SL + Focal Loss | 0,478 |
| | SL + Focal Loss + Tversky Loss | 0,493 |
| Eingabegröße, Skalierung (PSP-net mit ResNet-18, Softmax Loss + Focal Loss) | 480, 450 | 0,478 |
| | 960, 450 | 0,376 |
| | 480, 900 | 0,437 |
| | 960, 900 | 0,463 |
| Skalierung (DeepLabV3+ mit Xception-Net, Eingabegröße=480) | 450 | 0,493 |
| | 900 | 0,534 |

4.4.5 Visualisierung der Ergebnisse auf Asphalt Daten

In diesem Abschnitt werden die visualisierten Netzwerkausgaben des finalen Modells aus dem Projekt ASFaLT neben den visualisierten Labels präsentiert, damit ein visueller Eindruck über die Ergebnisse geliefert werden kann. Die Bewertungsmaße dieses Modells werden neben den Bewertungsmaßen des besten Modells aus dem vorhergehenden Abschnitt in der folgenden Tabelle aufgelistet:

Tabelle 9: Ergebnisse auf den verschiedenen Datensplits

| Datensatzsplit | mIoU | Occurrence-Score (Micro) | Occurrence-Score (Macro) |
|-----------------------|-------|--------------------------|--------------------------|
| Val. (2020_06) | 0.479 | 0.789 | 0.376 |
| Test (2020_06) | 0.405 | 0.742 | 0.324 |
| Val. (2021_04) | 0.537 | 0.836 | 0.519 |

In den nachfolgenden Abbildungen werden die visualisierten Ergebnisse auf den Testdaten des Datensatzes 2021_04 gezeigt.



(a) Label



(b) Ausgabe



(a) Label



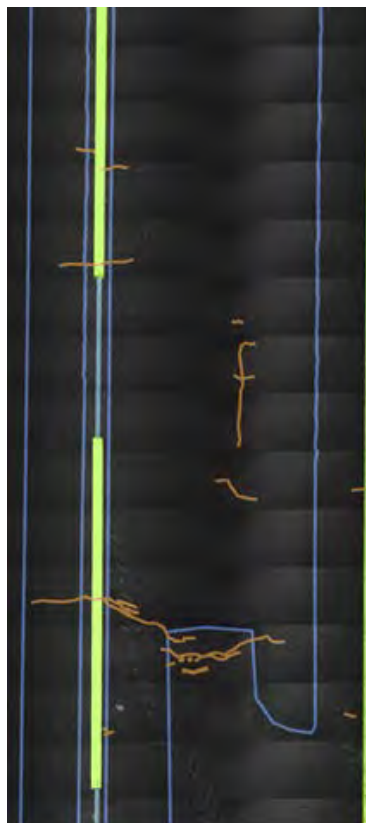
(b) Ausgabe



(a) Label



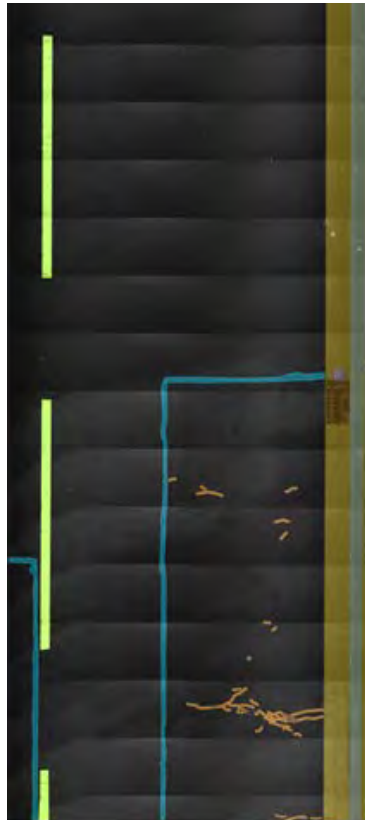
(b) Ausgabe



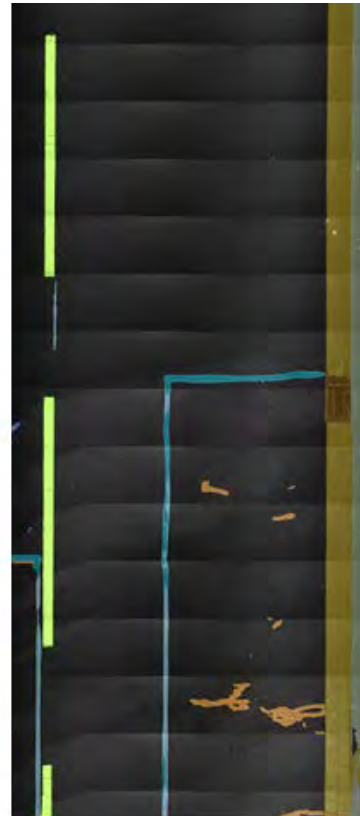
(a) Label



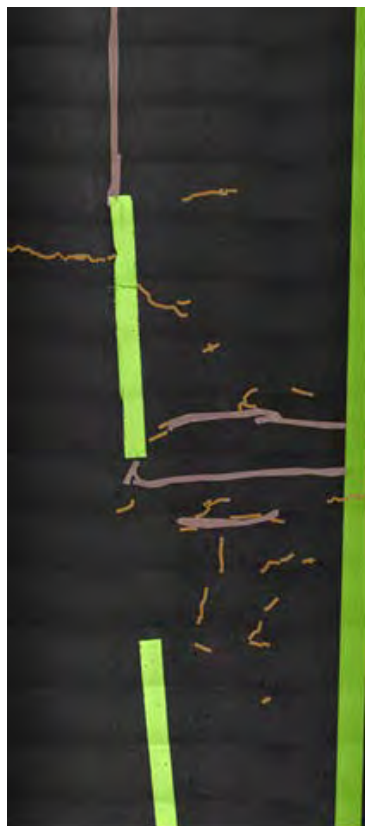
(b) Ausgabe



(a) Label



(b) Ausgabe



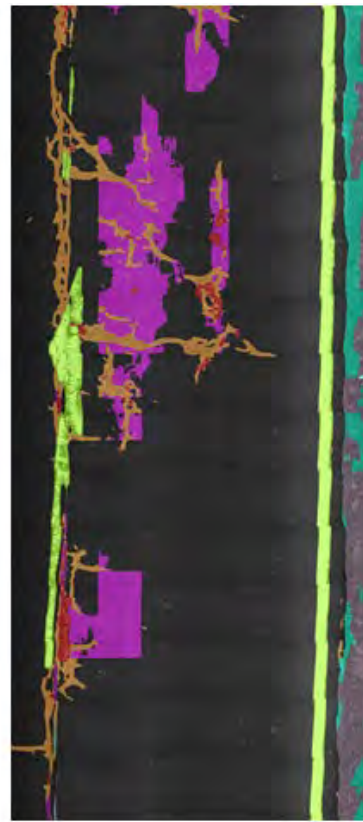
(a) Label



(b) Ausgabe



(a) Label



(b) Ausgabe



(a) Label



(b) Ausgabe

4.5 Segmentierungsergebnisse auf Beton

Die Schadstellen- und Objektdetektion auf Asphaltoberflächen wurde im Rahmen des Projekts zuerst bearbeitet, um gut geeignete Architekturen und Trainingsstrategien ableiten zu können. Für Betonoberflächen sollen die auf Asphalt gemachten Erfahrungen übertragen werden, sodass der Zeitaufwand für die Entwicklung gut geeigneter Klassifikatoren auf Betonoberflächen deutlich weniger Zeit in Anspruch nehmen sollte. Für das Training der Netzwerke auf der Oberflächenart Beton kommen die gleichen Netzwerkarchitekturen wie auf Asphaltoberflächen zum Einsatz. Entsprechend bildet ein Netzwerk bestehend aus einem Resnet18 Encoder, PPM Kontextmodul und PSPNet Decoder die Basis für die hier gezeigten Ergebnisse.



Abbildung 17: Farbpalette für die Visualisierung von Schäden und Objekten auf Betonoberflächen.

4.5.1 Quantitative Bewertung

Die Auswertung des Betondetektors erfolgt, anders als bei Bewertung des Asphalt-Detektors, ausschließlich auf den Validierungsdaten. Aufgrund der geringeren Anzahl an Daten und der Verteilung der Daten war die Abspaltung eines Testdatensatzes für die Beton-Daten nicht möglich. Dies gilt es bei der Bewertung der Ergebnisse zu berücksichtigen.

Für die Visualisierung von Ergebnisbildern wird ein zur Asphalt-Auswertung identisches Vorgehen gewählt. Da sich die Klassen auf Beton-Oberflächen jedoch unterscheiden, kommt in diesem Abschnitt die in Abbildung 17 gegebene Farbpalette zur Anwendung.

Bei dem Training der Beton-Detektoren zeigt sich, dass die Anzahl an Trainingsdaten für einige Schäden und Objekttypen noch größer ausfallen könnte. Dies zeigt sich beispielsweise daran, dass die Ergebnisse von Detektoren, die mit gleichen Trainingsparametern

trainiert wurden und somit nur in der Initialisierung variieren, deutlich mehr streuen, als dies bei den Asphalt-Detektoren der Fall ist. Um dies weiter zu untersuchen, müsste perspektivisch (analog zum Vorgehen bei Asphalt) noch ein Testdatensatz aufgebaut werden, um den Unterschied der Klassifikation auf Validierungsdaten und auf Testdaten analysieren zu können.

Für den besten im Projekt trainierten Beton-Detektor ergeben sich die in Tabelle 10 gegebenen Bewertungsmetriken.

Tabelle 10: Bewertungsmaße für die mit dem Beton-Detektor erzielten Ergebnisse.

| mIoU | Occurrence-Score (Micro) | Occurrence-Score (Macro) |
|-------|--------------------------|--------------------------|
| 0.407 | 0.93 | 0.48 |

Auf den ersten Blick scheint der Beton-Detektor auf einem ähnlichen Niveau wie der Asphalt-Detektor zu arbeiten und erreicht für den Micro Occurrence sogar besserer Werte. Bei genauerer Betrachtung zeigt sich aber, dass die Werte vor allem auf zwei Umstände zurückzuführen sind. Zum einen wird der Beton-Detektor auf deutlich weniger Klassen trainiert, so dass die gute Detektion von einfachen Klassen (Markierung, Plattengrenze, Normal) stärker ins Gewicht fällt. Zum anderen sind auf den Betondaten generell weniger Schäden vertreten, was sich in dem hohen Micro-Score widerspiegelt.

4.5.2 Qualitative Bewertung

Bei der qualitativen Analyse der Validierungsdaten zeigt sich, dass das Netzwerk bei vielen Bildern bereits eine recht gute Detektion erzielen kann. So werden Risse und auch Asphaltbereiche auf vielen Bildern korrekt lokalisiert, so dass eine große Übereinstimmung mit den gelabelten Bildern erzielt werden kann (Abbildung 18, Abbildung 19), was sich auch bereits an dem hohen micro-Occurrence-Score zeigt.

Es zeigen sich jedoch bei einigen Bildern auch noch systematische Probleme. Eines dieser Probleme ist beispielsweise die Unterscheidung zwischen Rissen und Ausbrüchen, die in vielen Fällen fließend ist. In Abbildung 20 ist beispielsweise ein Fall dargestellt, in dem die an der Flickstelle entstandenen Schäden von dem Detektor der Klasse Riss zugeordnet werden. Derartige Probleme sollten sich allerdings durch die Nutzung weiterer Trainingsdaten beseitigen lassen, so dass von dem Detektor während des Trainings mehr Beispiele für die korrekte Unterscheidung der beiden Klassen genutzt werden können. Ein weiteres

Problem zeigt sich in der teilweise schlechten Detektion von sehr feinen Rissen (siehe Abbildung 21). Dabei zeigt sich bereits in den Trainingsdaten, dass das Label Riss sehr spezifisch für bestimmte Oberflächen vergeben wurde. Manche Betonoberflächen weisen grundsätzlich eine gewisse feine Rissstruktur auf. Risse wurden auf diesen Oberflächen erst bei einer deutlichen Ausprägung gelabelt. Auf anderen Betonoberflächen werden jedoch Strukturen, die den feinen Rissstrukturen sehr ähnlich sind, bereits als Risse gelabelt. Dies erschwert das Training für den Detektor deutlich und in der Summe scheint es auf den Trainingsdaten damit besser zu sein, feine Risse nicht als solche zu detektieren.

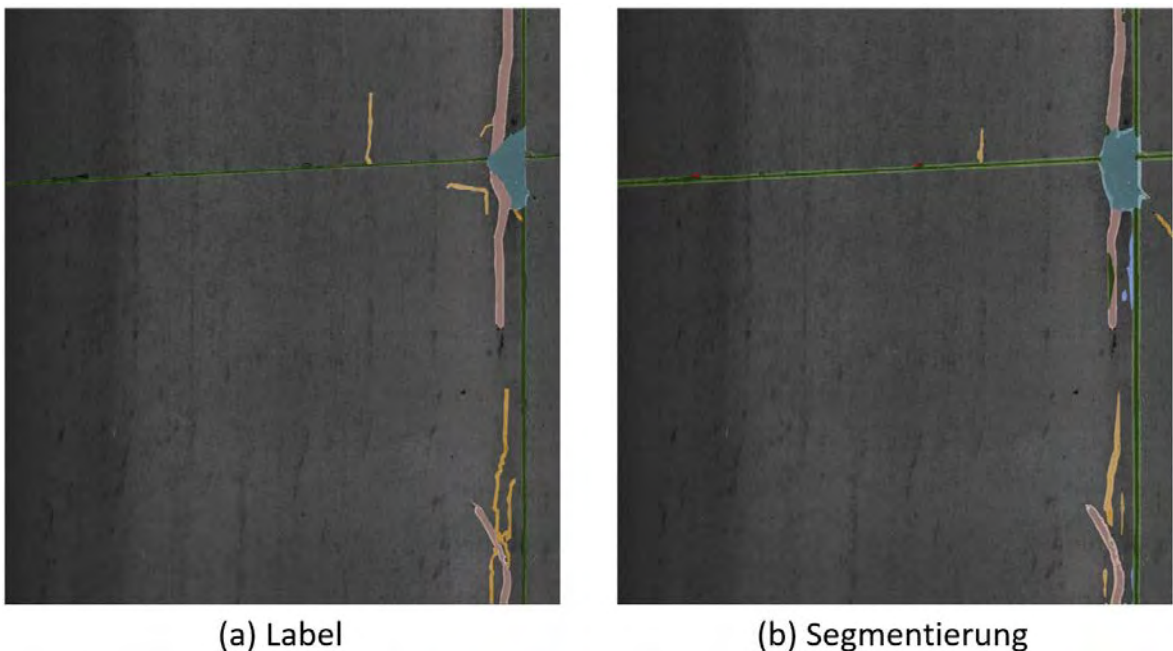
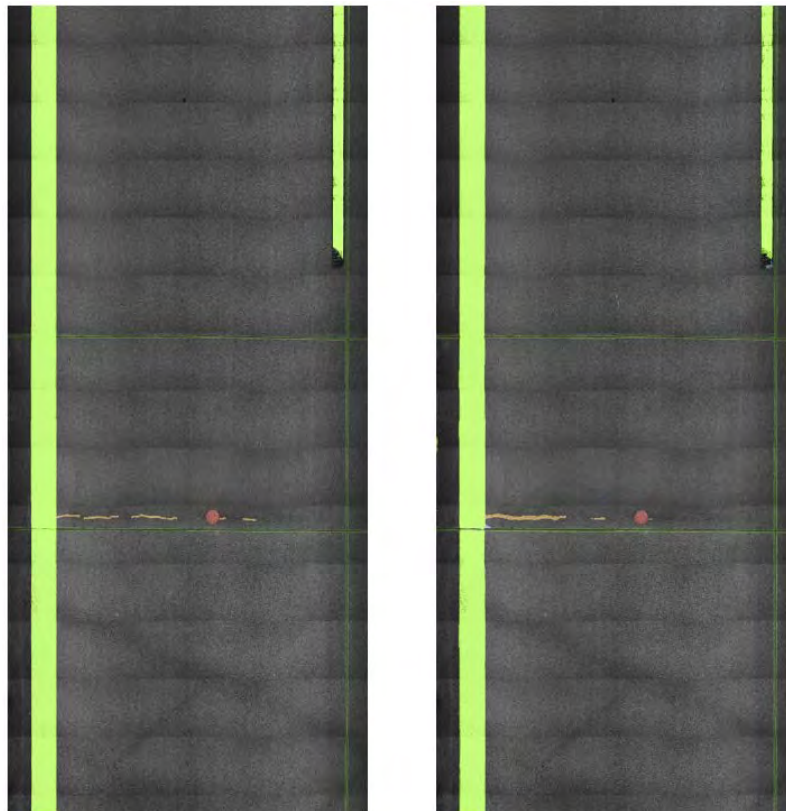


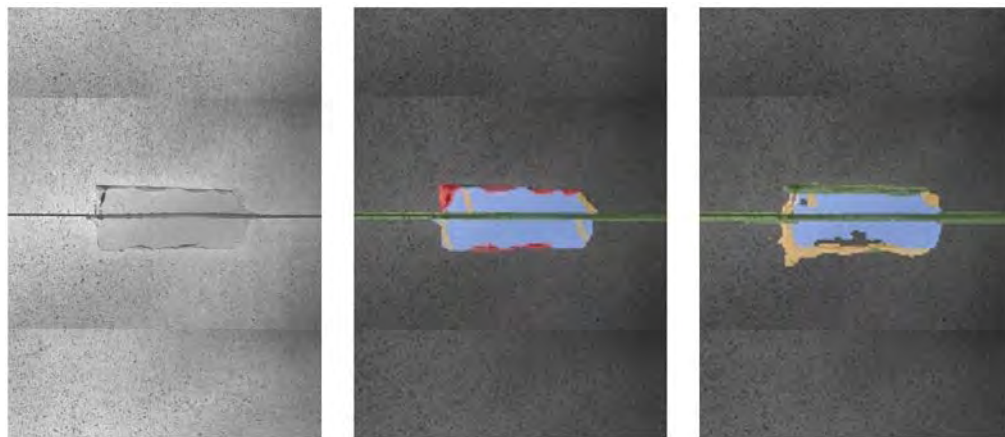
Abbildung 18: Beispiel für gute Segmentierung. Gelabeltes Oberflächenbild (a) und Segmentierung (b) stimmen in den wesentlichen Grundzügen überein. Rissen werden überwiegend ähnlich verortet und auch Asphaltbereiche gut erkannt.



(a) Label

(b) Segmentierung

Abbildung 19: Beispiel für gute Segmentierung. Gelabeltes Oberflächenbild (a) und Segmentierung (b) stimmen gut überein. Rissen und Ausbrüche werden fast identisch lokalisiert.



(a) Oberflächenbild

(b) Label

(c) Segmentierung

Abbildung 20: Fehlerhafte Detektion von Ausbrüchen als Risse.

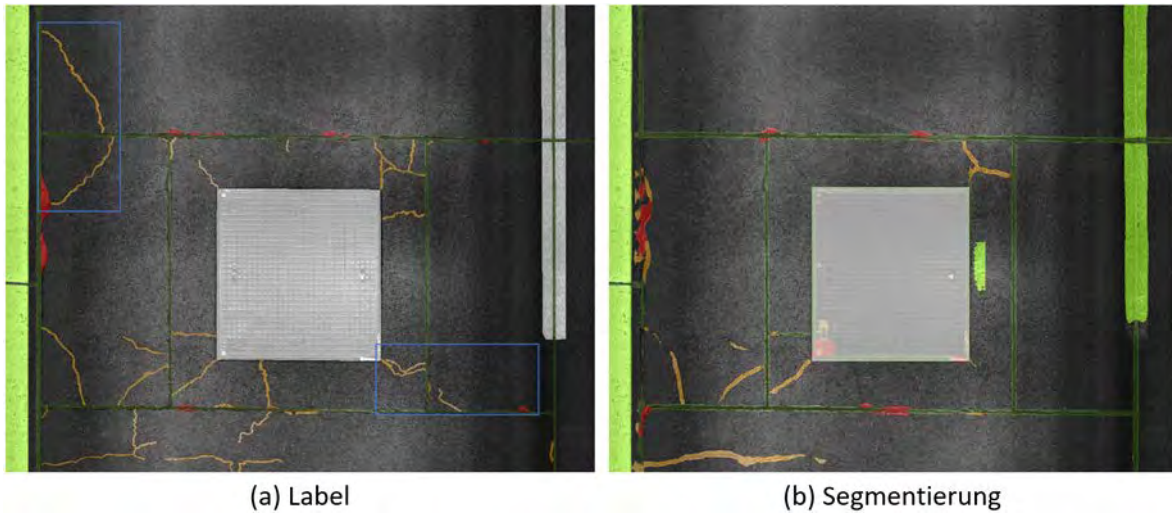


Abbildung 21: Sehr feine Risse werden von dem Detektor in einigen Fällen nicht erkannt. Hierbei ist insbesondere auf die mit einem blauen Rechteck markierten Bereiche zu achten.

Auf Grundlage der zur Verfügung stehenden Daten konnte bereits ein Detektor trainiert werden, der eingeschränkt für die Detektion von Schäden auf Betonoberflächen geeignet ist. Wie gut diese Detektion in der Praxis ist, soll unter Abschnitt 9 noch einmal genauer untersucht werden.

4.6 Vergleich mit menschlicher Segmentierungsleistung

Die von der Automatisierung erzielten Bewertungsmaße fallen auf den ersten Blick gering aus. Um die erzielten Ergebnisse für die Schadenssegmentierung besser bewerten zu können, erfolgte auch eine Auswertung von 14 Oberflächenbildern (AIT-Standard, 4m x 3m, Asphaltoberfläche) durch jeweils 3 Experten für die Bewertung von Oberflächenschäden. Ziel dieser Untersuchung war das Erstellen einer Referenz, um auch die Streuung der menschlichen Bewertung der automatisierten Bewertung gegenüberstellen zu können.

4.6.1 Methodik

Alle 14 Oberflächenbilder wurden von den 3 Experten unabhängig voneinander gelabelt. Dabei wurde ein zu dem Erstellen von Trainingsdaten identisches Labeling verwendet (Pixelgenau, 28 Schadenklassen).

Anschließend wurde jeweils eine der Personen als „Ground Truth“, also als Referenz, herangezogen und die Ergebnisse von den beiden verbleibenden Experten und die Ergebnisse der KI-Auswertung gegen diese Ground Truth verglichen. Auf diese Weise ergeben sich 3 Testdurchläufe, da jeder Experte einmal als Referenz angesehen wird.

4.6.2 Ergebnisse

Die Ergebnisse für die Gegenüberstellung der von den Experten und der KI erzielten Segmentierung ist in Abbildung 22 dargestellt.

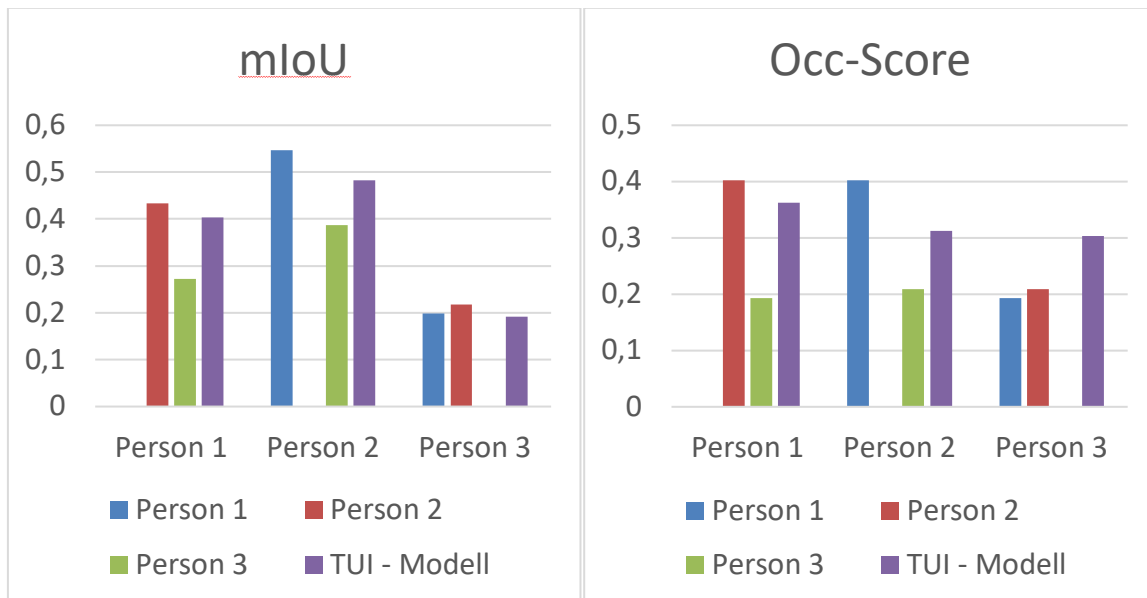


Abbildung 22: Gegenüberstellung der Segmentierungsergebnisse von 3 Experten und der KI. Die Ergebnisse für die mIoU (links) und den Occ-Score (rechts) sind jeweils in drei Gruppen unterteilt. Jeder Experte wurde für eine dieser Gruppen als Ground Truth angenommen.

Es ist gut zu erkennen, dass die KI sich sehr gut in die erzielten Ergebnisse der Experten eingliedert. In allen Test liefert die KI etwa das mittlere Ergebnis zwischen den Scores der Experten. Damit bildet die KI eine gute Mittlung über alle Experten ab.

In der Summe lässt sich daher behaupten, dass die KI trotz der etwas gering anmutenden Bewertungsmaße durchaus eine Segmentierung auf menschlichem Niveau erzielt.

4.7 Fazit

Mit dem im Rahmen des Projekts ASFaLT gelabelten Bildern konnten für Asphalt- und Betonoberflächen getrennt trainierte Detektoren erzeugt werden, welche in der Lage dazu sind, eine Schadens- und Objektdetektion auf den jeweiligen Oberflächen durchzuführen. Durch den Vergleich mit mehreren manuell gelabelten Testdaten konnte gezeigt werden, dass die Detektion auf Asphaltoberflächen dabei schon durchaus mit der manuellen Bewertung vergleichbar ist. Ein solcher Vergleich für Betonoberflächen konnte in der Projektlaufzeit nicht mehr durchgeführt werden. Es wurden bei dem Betondetektor jedoch ein paar systematische Probleme ausgemacht, die vermutlich hierbei zu einer schlechteren Bewertung führen.

5 Qualitätsmaß für die Detektionsgenauigkeit

Die von den Neuronalen Netzwerken ermittelten Segmentierungsergebnisse lassen in der Regel nur einen eingeschränkten Rückschluss auf die Sicherheit der von dem Netzwerk getroffenen Entscheidungen zu. Die oft verbreitet als Sicherheit interpretierte Softmax-Ausgabe (falls vorhanden) ist in vielen Fällen kein günstiges Maß, um die Sicherheit einer Entscheidung abzuleiten. Im Folgenden sollen daher typische Verfahren für die Ermittlung von Unsicherheit vorgestellt werden, welche die Grundlage für ein Qualitätsmaß bilden. Anschließend werden diese zunächst auf synthetischen Daten und anschließend auf Projektdaten angewendet und überprüft, wie Unsicherheiten im Rahmen des Projekts gewinnbringend eingesetzt werden können.

Die Grundlagen der Verfahren und die Tests auf synthetischen Daten sind der Masterarbeit von Benedict Stephan [Stephan 2019] entnommen, welche im Rahmen des Projekts entstanden ist.

5.1 Arten von Unsicherheiten

Im Kontext der Klassifikation und Regression mit neuronalen Netzwerken können laut [Kureghian 2009] zwei grundsätzliche Arten von Unsicherheiten unterschieden werden. Dies ist zum einen die Unsicherheit, welche sich durch Hinzunahme weiterer Datenpunkte während des Trainings verringern lässt. Diese Unsicherheit entsteht, wenn die geschätzten Modellparameter des Netzwerks nicht korrekt sind, und schlägt sich vor allem in Bereichen des Eingaberaums nieder, in denen keine Trainingsdaten vorhanden sind. Diese Art der Unsicherheit bezieht sich direkt auf das Modell und wird folglich auch als Modellunsicherheit bezeichnet.

Die zweite Quelle von Unsicherheiten entsteht beispielsweise, wenn Trainingsdaten widersprüchlich sind und dadurch die korrekte Schätzung der Modellparameter verhindern. Da sich diese Art von Unsicherheit auf die Daten zurückführen lässt, wird sie auch als Datenunsicherheit bezeichnet. Die unterschiedlichen Arten von Unsicherheiten sind in Abbildung 23 visualisiert.

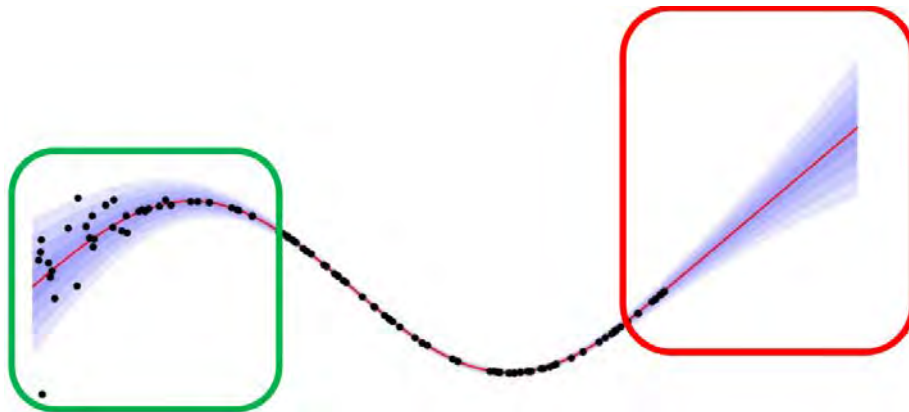


Abbildung 23: Visualisierung der zwei Haupttypen von Unsicherheiten [Stephan 2019]. Zu sehen ist eine Ausgabe eines hypothetischen Modells in rot. Die schwarzen Punkte stellen die Trainingspunkte für das Model dar. Der Anstieg der Unsicherheit (blau) am linken Rand (grüne Box) wird durch das vorhandene Rauschen in den Datenpunkten verursacht (Datenunsicherheit), während die Unsicherheit am rechten Rand (rote Box) entsteht, da hier keine Trainingsbeispiele vorhanden sind und diese Region für das Model somit unbekannt ist (Modellunsicherheit).

5.2 Methoden zur Bestimmung von Unsicherheiten

Die probabilistischen Methoden zur Schätzung von Unsicherheiten in Neuronalen Netzwerken lassen sich grundsätzlich in die in Abbildung 24 gegebene Taxonomie eingruppiieren. Im Folgenden soll eine knappe Einführung in die unterschiedlichen Verfahren gegeben werden, wobei für eine ausführliche Beschreibung auf [Stephan 2019] verwiesen wird. Abschließend erfolgt eine Einschätzung der Relevanz der Verfahren in Bezug auf das Projekt ASFaLT.

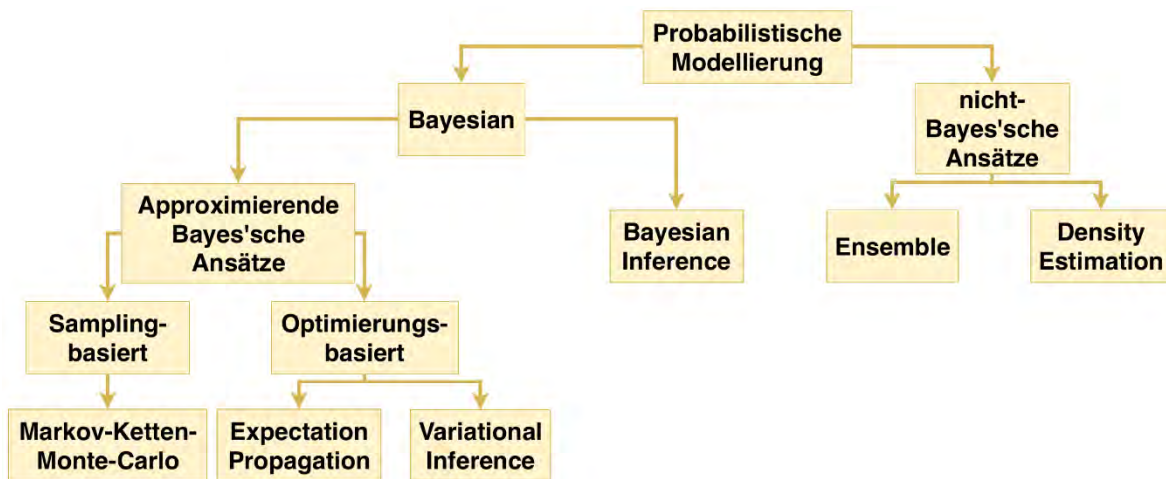


Abbildung 24: Grundlegende Einteilung der Ansätze in hierarchisch zusammenhängenden Gruppen [Stephan 2019]

5.2.1 Bayes'scher Ansatz

Bei dem Training eines Neuronalen Netzwerkes wird in der Regel eine Optimierung über den Parametern (Gewichten) θ eines Netzwerkes durchgeführt, so dass der Fehler des Netzwerkes über den Trainingsdaten minimal wird.

Seien $\mathbf{D} = \{(\mathbf{t}_i, \mathbf{x}_i)\}_{i=1}^M$ die Trainingsdaten des Netzwerkes mit den Eingaben \mathbf{x}_i und den Zielwerten \mathbf{t}_i , kann, im Sinne der Wahrscheinlichkeitstheorie, diese Optimierung als Maximierung einer Likelihood-Funktion $p(\mathbf{D}|\theta)$ über den Netzwerkparametern θ verstanden werden. Dabei wird eine Parameterkonfiguration θ für ein Netzwerk gesucht, so dass die Wahrscheinlichkeit maximal ist, dass die für das Training verwendeten Daten von dem Modell erzeugt wurden. Unter Anwendung der vollständigen Bayes'schen Formel lässt sich folgende Beziehung aufstellen:

$$p(\theta|\mathbf{D}) = \frac{p(\mathbf{D}|\theta)p(\theta)}{p(\mathbf{D})} = \frac{p(\theta)p(\mathbf{D}|\theta)}{\int p(\mathbf{D}|\theta)p(\theta)d\theta}$$

$p(\theta)$ beschreibt hierbei die sogenannte A-priori-Verteilung. Diese Verteilung beinhaltet Vermutungen oder Vorkenntnisse über die Modellparameter. $p(\theta|\mathbf{D})$ beschreibt hingegen die sogenannte A-posteriori-Verteilung. Diese ist die Schlussfolgerung auf Basis der A-priori-Verteilung und der beobachteten Daten \mathbf{D} .

$p(\mathbf{D})$ repräsentiert die Evidenz, die, unabhängig von den gewählten Parametern, beschreibt, wie gut ein Modell zu den Daten passt. Auf Basis diese Beziehung können auch Vorhersagen für eine beliebige Eingabe getroffen werden, wobei sogenannte Prädiktive Verteilungen entstehen, welche Aussagen über die Sicherheit einer Schätzung ermöglichen (Details in [Stephan 2019]). Als problematisch erweist sich dabei jedoch, dass für die Berechnung unter anderem das Integral über die A-posteriori-Verteilung gebildet werden muss. Für umfangreiche Datensätze oder Modelle mit vielen Parametern kann dieses Integral jedoch oft nicht mehr berechnet werden, was den Einsatz von approximierenden bayesschen Ansätzen erfordert.

5.2.2 Approximierende bayessche Ansätze

Ziel der approximierenden Ansätze ist die Abschätzung der A-posteriori-Verteilung ohne diese explizit analytisch zu berechnen. Dabei wird wiederum zwischen zwei unterschiedlichen Ansätzen, nämlich den Optimierungsbasierten-Ansätzen und den Sampling basierten-Ansätzen unterschieden.

Optimierungsbasierte Ansätze

Bei diesen Verfahren wird die eigentliche A-posteriori-Verteilung durch eine Ersatzverteilung (z.B. Normalverteilung) approximiert, indem die Divergenz zwischen beiden Verteilungen minimiert wird. Entsprechend des verwendeten Divergenzmaßes erfolgt noch einmal eine Untergliederung in zwei Gruppen. Verfahren, welche die Forward Kullback-Leibler-Divergenz nutzen, werden den Verfahren der Expectation Propagation zugeordnet, während Verfahren, die auf der Backward Kullback-Leibler-Divergenz basieren, den Verfahren der Variational Inference zugerechnet werden. Ein klassischer Vertreter der Expectation Propagation wird in [Minka 2004] beschrieben. Da bei diesem Verfahren jedoch für jeden Datenpunkt im Trainingsdatensatz ein eigener Likelihood-Term eingeführt wird, ist das Verfahren sehr speicher- und rechenintensiv. Die bei der Variational Inference eingesetzte Backward Kullback-Leibler-Divergenz lässt sich effizienter berechnen, wodurch diese Verfahrensgruppe eine größere Aufmerksamkeit erfährt. Ein sehr bekannter Vertreter aus dieser Gruppe von Verfahren ist das in [Gal 2016b] vorgestellte Monte-Carlo Dropout (MC-Dropout). Hierbei wird das Training eines Netzwerkes mit Dropout-Layern ganz normal durchgeführt. Allerdings wird anschließend auch während der Inferenz Dropout angewendet, um die A-posteriori-Verteilung durch unterschiedliche Dropout-Masken approximieren zu können. Die Schätzung der Unsicherheit kann dabei über die Varianz der unterschiedlichen Netzwerkausgaben erfolgen.

Samplingbasierte Ansätze

Im Vergleich zu den optimierungsbasierten Ansätzen wird bei dieser Gruppe an Verfahren keine spezielle Verteilungsart angenommen. Vielmehr wird versucht die A-posteriori-Verteilung zielgerichtet derart zu sampeln, dass die Samples die eigentliche Verteilung repräsentieren. Dieser Schritt erfolgt durch Markov Chain Monte Carlo (MCMC) Methoden [Hastings 1970, Gelfand 1990]. Ein wesentlicher Nachteil dieser Verfahrensgruppe ist jedoch der große Ressourcen-Verbrauch der Markov-Kette. Jedes einzelne Sample der Kette wird durch den kompletten Parametersatz des Netzwerkes und damit durch Millionen von Gewichten repräsentiert. Weiterhin dauert es in der Regel relativ lange bis die Markov-Kette auf die A-posteriori-Verteilung konvergiert. Ein Beispiel aus dieser Verfahrensgruppe wird durch den No-U-Turn Samples (NUTS) in [Hoffman 2014] beschrieben, welcher mit wenigen Parametern eine robuste Schätzung der Verteilung ermöglicht.

5.2.3 Nicht-bayessche Ansätze

Nicht-bayessche Verfahren basieren nicht auf den in der Bayes'schen Formel (5.2.1) definierten Eigenschaften. Einen klassischen Vertreter aus dieser Gruppe stellen die Verfahren der Ensembles dar, welche Unsicherheiten über den Einsatz von mehreren Netzwerkmodellen abschätzen. Ein wesentliches Problem besteht dabei darin, die Modelle des Ensembles so zu trainieren, dass sie ausreichend divers sind. Dazu finden sich in [Lakshminarayanan 2017] Untersuchungen zu verschiedenen Techniken, die diese Diversität sicherstellen zu können.

Eine weitere Möglichkeit der Schätzung der Unsicherheit besteht darin, die Ausgabe eines Netzwerkes direkt als parametrisierte Verteilung zu definieren. So kann das Netzwerk beispielsweise derart trainiert werden, dass es nicht nur den Mittelwert μ sondern auch die Varianz σ als Ausgabe schätzt. Hierbei ist jedoch zu beachten, dass diese Schätzung nur über die Trainingsdaten erfolgt und mit diesem Vorgehen nur die Datenunsicherheit abgeschätzt werden kann. Ein Verfahren, welches auf diese Weise die Datensicherheit schätzt, wird beispielsweise in [Kendall 2017] vorgestellt.

Die für Klassifikationsprobleme oft genutzt Softmax-Ausgabe normiert die Summe aller Ausgabeneuronen auf 1 und kann daher als Verteilung interpretiert werden. Auch wenn diese Verteilung keinen theoretischen Zusammenhang mit der prädiktiven Verteilung besitzt, wird sie häufig genutzt, um Aussagen über die Sicherheit von Modellaussagen zu treffen. Dabei wurde in [Guo 2017] gezeigt, dass modernere Netzwerkarchitekturen generell dazu tendieren, bei einzelnen Neuronen Ausgaben nahe 1 zu erzeugen und somit „sehr sichere“ Aussagen zu treffen. Daher wurde in der Veröffentlichung gezeigt, wie die Ausgabe von Netzwerken, beispielsweise durch den Einsatz von Temperature Scaling, derart kalibriert werden kann, dass sich die Ausgaben den tatsächlichen Unsicherheiten annähern.

5.2.4 Fazit

Während sich in der Literatur eine Vielzahl unterschiedlicher Verfahren für die Schätzung von Unsicherheiten finden lassen, sind, insbesondere aufgrund ihrer Ressourcenanforderungen, nicht alle Verfahren für den Einsatz im Projekt ASFaLT geeignet. Während beispielsweise samplingbasierte Ansätze eine qualitativ gute Schätzung von Unsicherheiten versprechen, sind sie für die Anwendung im Projekt ASFaLT ungeeignet, da die Berechnung viel zu zeitaufwendig wäre. Ihr Einsatz beschränkt sich im Projekt daher auf die Bereitstellung einer Referenz für Untersuchungen auf synthetischen Daten. Relativ ressour-

censhonende Verfahren, welche sich auch für das Projekt ASFaLT eignen, stellen hingegen das Monte-Carlo Dropout und die Verwendung von Ensembles dar. Weiterhin ist auch die Schätzung der Datenunsicherheit durch den Ansatz nach Kendall im Projekt umsetzbar und wird im Folgenden näher untersucht.

5.3 Anwendung auf synthetischen Daten

Um die Eignung und Eigenschaften der vorgestellten Verfahren näher untersuchen zu können, erfolgte zunächst ein Test auf synthetischen Daten.

Auf diese Weise können die Rahmenbedingungen fest definiert und die Ergebnisse besser interpretiert werden.

5.3.1 Datengrundlage

Für die Tests auf synthetischen Daten wurde ein einfaches Regressionsproblem gewählt, welches einen periodischen Verlauf aufweist und somit die Möglichkeit bietet, zu untersuchen, wie sich die Verfahren verhalten, wenn sich der Abstand zwischen bekannten und unbekanntem Daten vergrößert. Die Wahl fiel auf die Kosinusfunktion, welche mit einem Rauschterm ε beaufschlagt wurde:

$$y = \cos(x) + \varepsilon(x)$$

Um sowohl die Auswirkungen von Daten- als auch von Modellunsicherheit mit dieser Funktion simulieren zu können, wurden zusätzlich folgende Randbedingungen definiert:

- Für das Training wurden Daten aus dem Intervall -4π bis $+4\pi$ verwendet, wobei die Datendichte von -4π bis $+4\pi$ stetig erhöht wurde.
- Für die Untersuchung von Modellsicherheit wurden Testdaten aus den in Tabelle 11 gegebenen Bereichen gezogen.
- Für die Untersuchung der Datenunsicherheit wurde die Trainings mit einem varianzhomogenen Rauschen $\varepsilon(x) \sim \mathcal{N}(0,0.1)$ beaufschlagt.

Eine Visualisierung der resultierenden Daten ist in Abbildung 25 dargestellt.

Tabelle 11: Beschreibung der Eingabebereiche. In dieser Tabelle sind die einzelnen Bereiche definiert. Dafür wird für jeden Bereich das Intervall, der verwendete Bezeichner und eine Kurzbeschreibung angegeben.

| Intervall | Bezeichner | Beschreibung |
|------------------|--------------|---|
| $[-6\pi, -5\pi]$ | G_weit_außen | Unbekannt, weit entfernt |
| $[-5\pi, -4\pi]$ | G_außen | Unbekannt, direkt an bekannten Daten mit geringer Dichte angrenzend |
| $[-4\pi, 0]$ | G_innen | Bekannt, geringe Datendichte |
| $[0, 4\pi]$ | H_innen | Bekannt, höhere Datendichte |
| $[4\pi, 5\pi]$ | H_außen | Unbekannt, direkt an bekannten Daten mit hoher Dichte angrenzend |
| $[5\pi, 6\pi]$ | H_weit_außen | Unbekannt, weit entfernt |

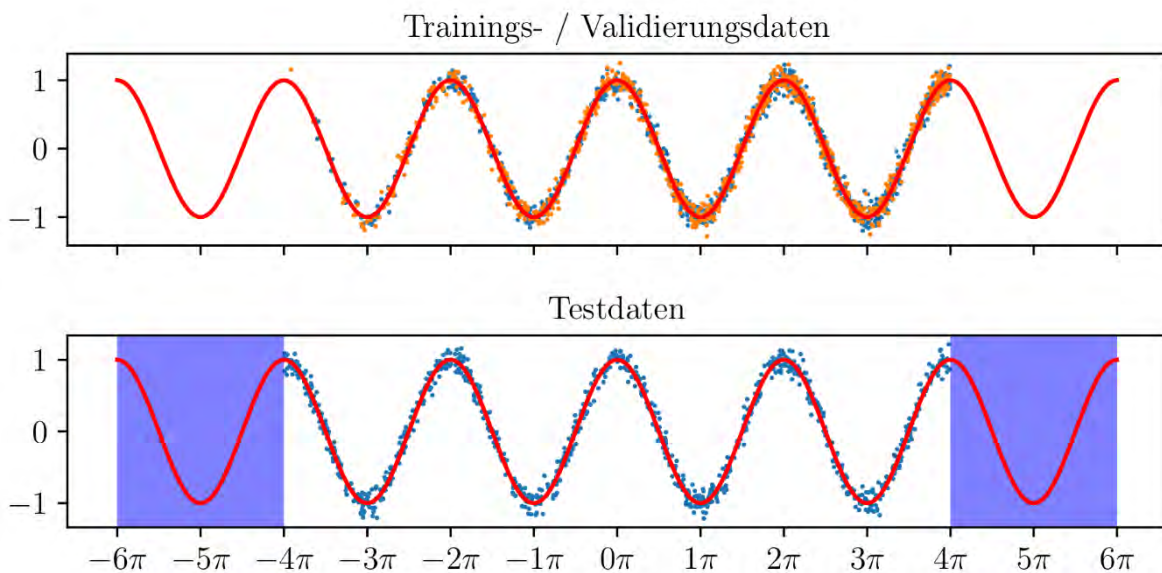


Abbildung 25: Streudiagramm des synthetischen Datensatzes [Stephan 2019]. Die obere Abbildung zeigt die Trainings- (blau) und Validierungsdaten (orange). Die untere Abbildung zeigt die Testdaten, auf denen die Modelle ausgewertet werden. Die blauen Bereiche repräsentieren dabei Bereiche, aus denen unbekannte Daten gezogen wurden. Die rote Linie zeigt die zugrundeliegende Kosinusfunktion.

5.3.2 Netzwerkmodelle

Für die Experimente wurden Netzwerke mit zwei Hiddenschichten und ReLU-Aktivierung verwendet, wobei die Anzahl der Neuronen in den Experimenten jeweils variiert wurde. Für die Schätzung der Modellunsicherheit wurden die Netzwerke mit unterschiedlichen Raten an Dropout in den Hiddenschichten trainiert (Monte-Carlo Dropout). Außerdem wurden zwei unterschiedliche Ausgabeschichten untersucht. Zum einen wurden Modelle mit nur einem Ausgabeneuron evaluiert, welche dadurch nur in der Lage sind Modellunsicherheiten zu

schätzen und nachfolgend als N-Modelle (Normal) bezeichnet werden. Zum anderen wurden Modelle mit zwei Ausgabeneuronen nach [Kendall 2017] trainiert, welche in der Lage sind sowohl die Modell- als auch die Datenunsicherheit zu schätzen und nachfolgend als K-Modelle (Kendall) bezeichnet werden. Als Referenz für die Schätzung der Unsicherheit wurde der No-U-Turn Sampler (NUTS) trainiert, da er mit der gewählten Problemstellung in annehmbarer Zeit trainiert werden konnte und auf Grund der Methodik eine sehr gute Schätzung der Unsicherheit ermöglichen sollte. Weitere Details zu den Trainingsparametern und dem genauen Ablauf der Experimente können [Stephan 2019] entnommen werden.

5.3.3 Erkennung von unbekanntem Daten

Ein wesentliches Ziel der Nutzung von Unsicherheiten im Projekt ASFaLT ist die Detektion von Situationen, in denen das Netzwerk auf Daten arbeitet, die sich so stark von den Trainingsdaten unterscheiden, dass bei der Anwendung Fehler zu erwarten sind und die Ergebnisse kontrolliert werden sollten. Es wurde daher untersucht, wie gut eine solche Situation durch die vorgestellten Methoden auf den synthetischen Daten detektiert werden kann. In der Abbildung 26 ist eine Übersicht der geschätzten Modellunsicherheiten auf den unterschiedlichen Bereichen für N-Modelle und K-Modelle abgebildet. Entlang der x-Achse sind, neben dem NUTS-Modell die Modellgrößen abgetragen. Die Balken beschreiben die mittlere Modellunsicherheit auf den jeweiligen Bereichen und die darüber liegenden Kurven beschreiben den dazugehörigen Mean Squared Error (MSE). Für jede Zeile in den Abbildungen wurde jeweils eine unterschiedliche Dropout-Rate verwendet.

N-Modelle

Für die Modellunsicherheiten ist zu erkennen, dass diese generell mit steigender Dropout-Rate für alle Bereiche und Datensätze ansteigen. Allerdings bilden die Modelle mit 32 Neuronen pro Schicht eine Ausnahme.

Für eine Dropout-Rate von $p = 0.2$ sind die Modellunsicherheiten auf den beiden unbekanntem Bereichen $G_{\text{außen}}$ und $G_{\text{weit_außen}}$ größer als bei einer Dropout-Rate von $p = 0.5$. Der Grund hierfür ist, dass kleine Modelle bei hoher Dropout-Rate eine einfache Funktion lernen (z.B. eine Gerade auf Höhe des Mittelwertes), während die Unsicherheit in den Parametern für diese Funktionen sehr gering ist. Anhand des Verlaufs des MSE in der Abbildung 26 ist zusätzlich erkennbar, dass die Modelle eine Funktion nahe der Geraden um den Mittelwert gelernt haben, da diese einen vergleichsweise geringen Fehler auf den unbekanntem Daten erzielen.

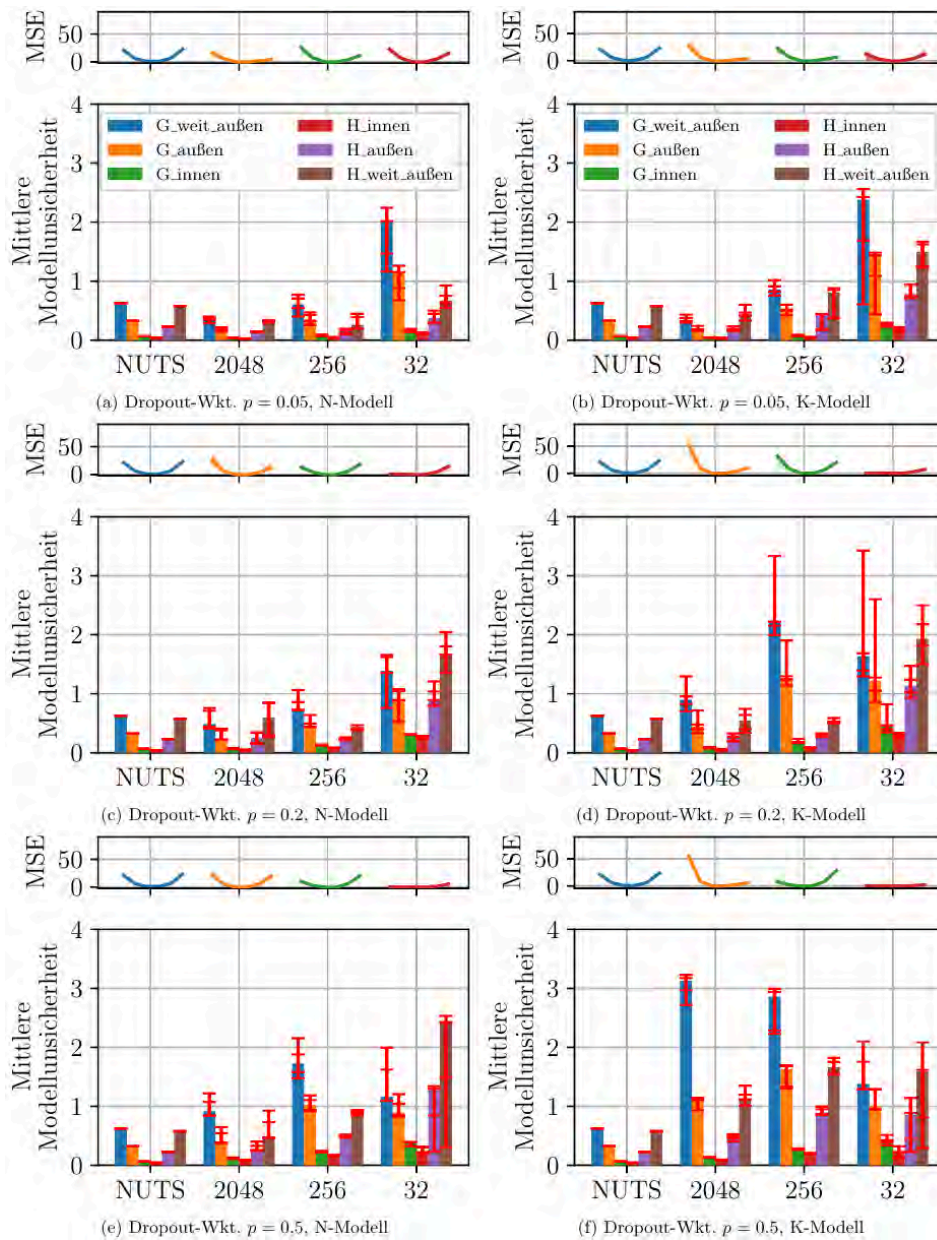


Abbildung 26: Balkendiagramm der mittleren Modellunsicherheiten für die verschiedenen Daten-Bereiche [Stephan 2019]. Links: N-Modelle, Rechts: K-Modelle.

Grundsätzlich zeigt sich, dass die erwarteten Eigenschaften für die Schätzung der Unsicherheit auf Basis des MC-Dropouts bei homogenem Datenrauschen gegeben sind. Die Unsicherheit steigt mit der Dropout-Rate an und die Dichte der Trainingsdaten scheint einen Einfluss auf die Unsicherheiten in den angrenzenden unbekanntem Bereichen zu haben.

K-Modelle

Auch für die K-Modelle scheinen die Modellunsicherheiten das gewünschte Verhalten für bekannte und unbekannte Bereiche zu zeigen. Generell ist die Modellunsicherheit für diesen Modelltyp allerdings größer als für die N-Modelle. Dies kann daran liegen, dass für die K-Modelle generell größere Modelle für die gleiche Qualität der Funktionsapproximation nötig sind. Da sich gezeigt hat, dass die Modellunsicherheit in einem gewissen Bereich für kleinere Modelle ansteigt, erklärt dies die erhöhte Unsicherheit im Vergleich zu den N-Modellen. Durch die Ähnlichkeit der Abbildungen für N-Modelle und K-Modelle, zeigt sich, dass eine Schätzung der Datenunsicherheit anhand des Verfahrens nach [Kendall 2017] keinen negativen Einfluss auf die Schätzung der Modellunsicherheit nach [Gal 2016b] hat.

5.3.4 Schätzung der Datenunsicherheit

Neben der Tatsache, dass sich die Schätzung der Datenunsicherheit kaum negativ auf die Schätzung der Modellunsicherheit auswirkt, muss natürlich auch untersucht werden, ob die Datenunsicherheit durch die K-Modelle überhaupt sinnvoll geschätzt werden kann. Da die synthetischen Daten künstlich verrauscht wurden, ist die tatsächliche Datenunsicherheit bekannt. Somit kann der Fehler zwischen der tatsächlichen und der geschätzten Datenunsicherheit einfach berechnet werden (Abbildung 27).

Leicht zu erkennen ist, dass die Modelle mit 32 Neuronen einen deutlich größeren Fehler als die restlichen Modelle erzeugen. In der Abbildung wurden die Fehler dieser Modelle abgeschnitten, um die restlichen Ergebnisse besser untersuchen zu können. Der große Fehler dieser Modelle ist darauf zurückzuführen, dass diese zu klein sind, um den Kosinus abbilden zu können. Die zusätzliche Schätzung der Datenunsicherheit verschärft dieses Problem, wodurch die Fehler sehr groß ausfallen.

Für die restlichen Modelle können zwei Zusammenhänge erkannt werden. Je größer das Modell, desto besser kann die Datenunsicherheit geschätzt werden und je größer die Dropout-Rate, desto größer ist der Fehler der geschätzten Datenunsicherheit. Diese Beobachtungen sind wieder analog zu dem Verhalten der Schätzung einer vorgegebenen Funktion, wie dem Kosinus, da die Schätzung der Datenunsicherheit der Abbildung einer Funktion gleicht.

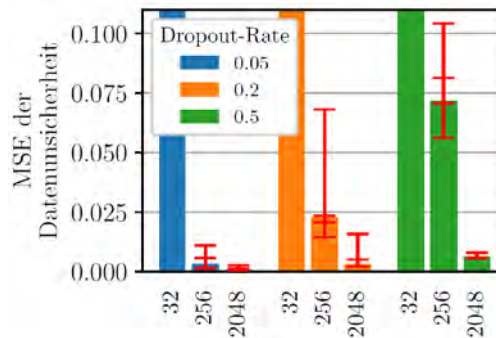


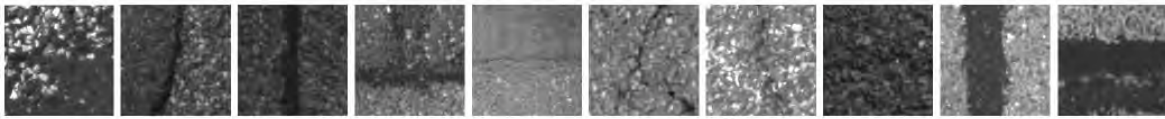
Abbildung 27: MSE der Schätzung der Datenunsicherheit [Stephan, 2019]. Die Abbildung zeigt den MSE der Schätzung der Datenunsicherheit für K-Modelle mit verschiedenen Dropout-Raten. An der x-Achse sind dabei die Modellgrößen abgetragen, während die verschiedenen Dropout-Raten farblich markiert sind.

5.4 Unsicherheiten bei der Anwendung auf Straßendaten

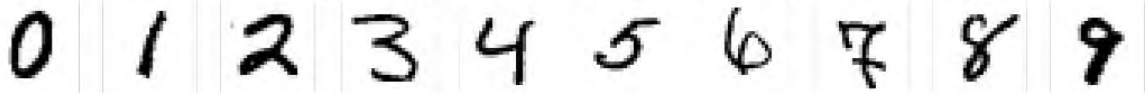
Nachdem die Tests zur Schätzung der Modell- und Datenunsicherheit auf den synthetischen Daten vielversprechende Ergebnisse zeigen, wurden im Rahmen des Projekts ASFaLT weitere Tests auf realen Bilddaten von Straßen durchgeführt. Da es bei den Labeldaten zu Verzögerungen kam, was auch das Training der im Projekt eingesetzten Segmentierungsansätze verzögert hat, wurden zunächst Tests auf Bilddaten des Projekts ASINVOS (GAPs Datensatz) und mit den im Rahmen des Projekts ASINVOS entwickelten, Patch-basierten Klassifikatoren durchgeführt. Dabei gilt zu beachten, dass der Umfang der Datenbasis damit deutlich geringer ausfällt, als es im Projekt ASFaLT angestrebt ist, und dass auch der Klassifikator nur auf Grundlage eines sehr eingeschränkten räumlichen Kontext Entscheidung trifft (der Klassifikator trifft jeweils für einen 160x160 Pixel großen Patch eine Klassenentscheidung, was nur einem Straßenausschnitt von wenigen Zentimetern entspricht).

5.4.1 Erkennung von unbekanntem Daten

Für die Untersuchung der Erkennung von unbekanntem Daten wurde zuerst ermittelt, wie gut die Methoden der Schätzung von Unsicherheiten geeignet sind, um Bilddaten zu detektieren, deren Verteilung sich deutlich von denen des ASFaLT Datensatzes unterscheidet. Dazu wurden 5 Netzwerke mit Dropout auf dem ASFaLT Datensatz trainiert und anschließend auf dem MNIST-Datensatz [LeCun 2010] und dem CIFAR-10 Datensatz [Krizhevsky 2009] angewendet. Auch wenn die Bilder dazu in Grauwertbilder konvertiert wurden, weisen die beiden Test-Datensätze deutlich anderen Strukturen als der ASFaLT Datensatz auf (Abbildung 28).



(a) Beispielbilder des GAPS Datensatzes



(b) Beispielbilder des MNIST Datensatzes mit handgeschriebenen Zahlen



(c) Beispielbilder des CIFAR10 Datensatzes mit den Klassen: Flugzeug, Automobil, Vogel, Katze, Hirsch, Hund, Frosch, Pferd, Schiff und Lastkraftwagen

Abbildung 28: Vergleich der Datensätze GAPS (a) MNIST (b) und CIFAR-10 (c) anhand von Beispieldaten.

Die Ergebnisse des Experiments werden in Abbildung 29 durch ROC-Kurven⁴ dargestellt. Es ist zu erkennen, dass die Detektion der unbekanntnen MNIST-Daten bei allen 5 Trainingsläufen fast perfekt möglich ist. Dies ist auch zu erwarten, da sich der einfache Aufbau der MNIST-Daten sehr stark von den ASFaLT Daten unterscheidet. Auf den CIFAR-10 Bildern zeigt sich ein etwas differenzierteres Bild. Zwar scheint die Unterscheidung der Daten in der Mehrzahl der Fälle möglich zu sein, schwankt jedoch deutlich bei unterschiedlichen Trainingsläufen. Erschwerend kommt hinzu, dass eine Vorhersage einer guten Detektion nicht über den F1-Wert⁵ des Modells während des Trainings (F1-Werte wurden auf Testdaten des ASFaLT Datensatzes berechnet) möglich ist. Es scheint daher erforderlich zu sein, eine Entscheidung über mehrere Modelle zu treffen, um Aussagen mit einer gewissen Sicherheit treffen zu können.

5.4.2 Modellunsicherheit

Nachdem die Tests auf dem MNIST und CIFAR-10 Datensätzen gezeigt haben, dass die Erkennung von unbekanntnen Daten auch auf Bilddaten grundsätzlich möglich ist, wurde im nächsten Schritt auf Bildern von Straßenoberflächen die Modellunsicherheit bestimmt. Dazu wurden wieder 5 Patch-basierte Netzwerkmodelle mit Dropout trainiert und anschließend der Mittelwert der Unsicherheitsschätzungen über alle 5 Modelle gebildet. Um den

⁴ ROC - receiver operating characteristic. ROC-Kurven die entlang der Diagonalen verlaufen, deuten auf einen Zufallsprozess hin. Im Idealfall erreicht ROC Kurve die obere Linke Ecke, was bedeuten würde, dass alle Beispiele von unbekanntnen Daten korrekt erkannt werden und auch die eigentlichen Daten des ASFaLT-Datensatzes über die Unsicherheit sicher identifiziert werden können.

⁵ Der F1-Wert ist das harmonische Mittel aus Genauigkeit (Precision) und Trefferquote (Recall) und wird sehr häufig für die Beurteilung der Qualität binärer Klassifikatoren eingesetzt.

Patch basierten Ansatz auch auf großen Eingabebildern visualisieren zu können, wurden die Bilder jeweils ausschnittsweise verarbeitet und anschließend zu einem Gesamtbild verrechnet. Auf diese Weise werden die Grundideen des Monte-Carlo Dropout und des Ensembleansatzes miteinander kombiniert.

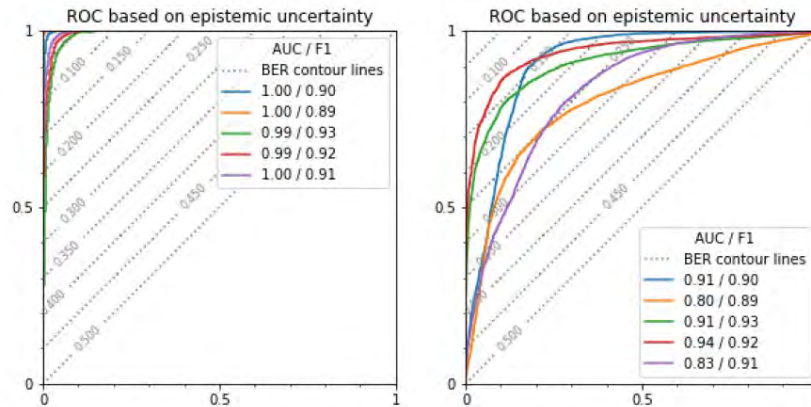


Abbildung 29: ROC-Kurven für die Unterscheidung von Test und Trainingsdatensätzen. Links: Detektion von MNIST-Bildern bei der Detektion auf ASFaLT. Rechts: Detektion von CIFAR-10 Daten bei Training auf ASFaLT.

Wie in Abbildung 30 beispielhaft dargestellt ist, werden Daten, welche im Training umfangreich repräsentiert waren, geringe Unsicherheiten zugewiesen. Bildausschnitte, welche im Training jedoch nur wenig vertreten waren oder deren Ausprägung nicht von den Trainingsdaten abgedeckt ist, erhalten hingegen eine hohe Unsicherheit (Abbildung 31). Bereits mit dem Patch-basierten Ansatz scheint die grundsätzliche Schätzung der Modellunsicherheit möglich zu sein. Damit zeigt sich bei den Untersuchungen das im Rahmen des Projekts ASFaLT gewünschte Verhalten, um Bereiche mit unsicheren Netzwerkentscheidungen detektieren zu können.

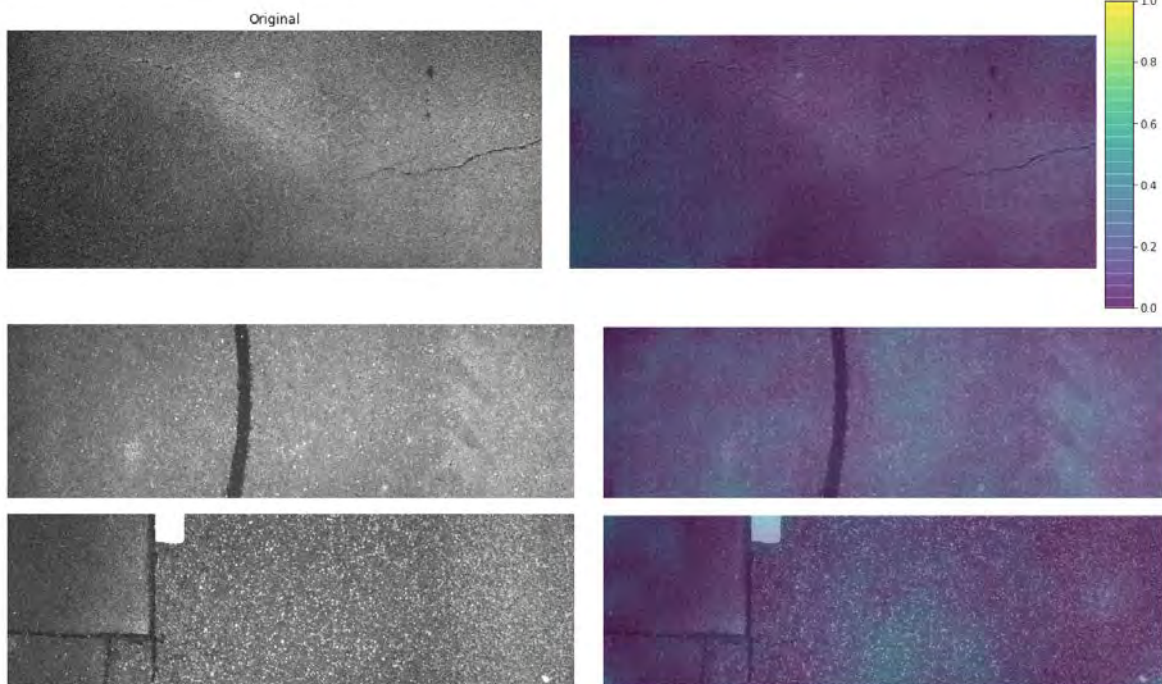


Abbildung 30: Schätzung der Modellunsicherheit auf Bilder mit großer Ähnlichkeit zu Trainingsdaten. Linke Spalte: originale Eingabebilder. Rechte Spalte: Eingabebilder mit Überlagerung der Modellunsicherheit (lila - geringe Unsicherheit, gelb - hohe Unsicherheit)

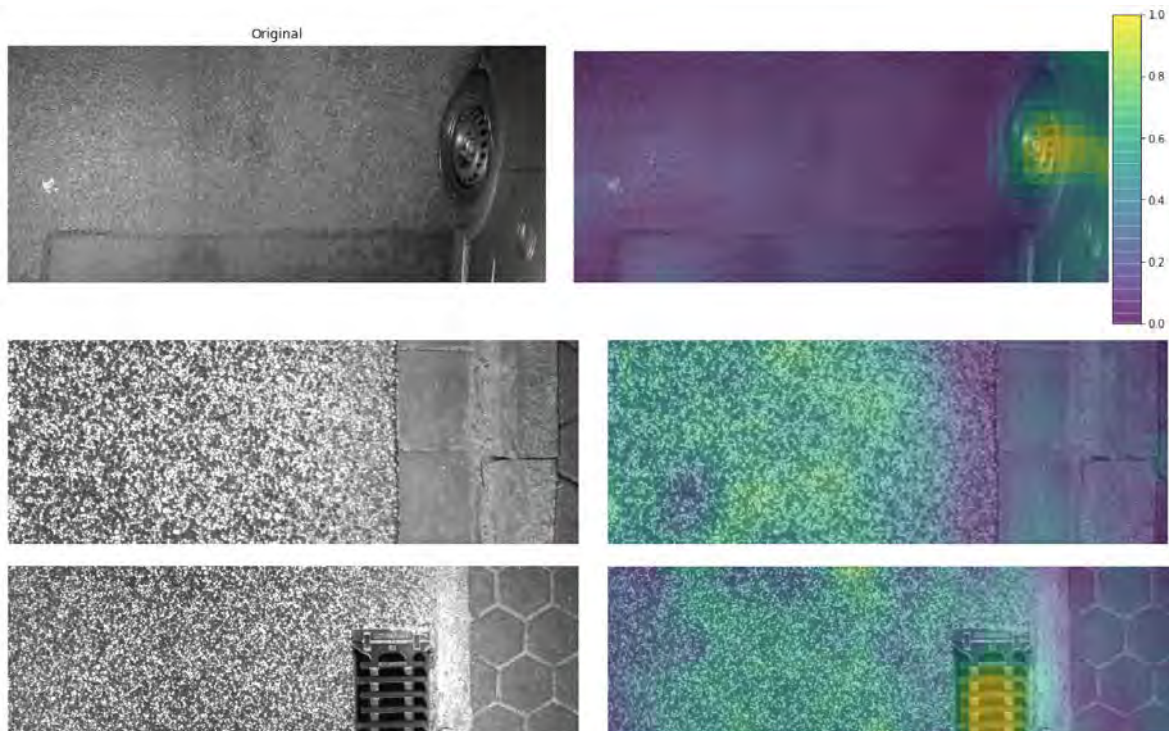


Abbildung 31: Schätzung der Modellunsicherheit auf Bildern, welche in Teilen nicht durch die Trainingsdaten repräsentiert sind. Oben - Da Objekte generell sehr wenig repräsentiert sind, ergeben sich hohe Unsicherheiten am Fahrzeug und besonders an den Reifen. Mitte/Unten - der Fahrbahnbelag erzeugt höhere Unsicherheit, da er im Training nicht repräsentiert war. Auch der Abfluss war in dieser Form nicht durch das Training abgedeckt. Linke Spalte: originale Eingabebilder. Rechte Spalte: Eingabebilder mit Überlagerung der Modellunsicherheit (lila - geringe Unsicherheit, gelb - hohe Unsicherheit)

5.5 *Zwischenfazit*

In synthetischen Test konnten verschiedene Methoden für die Schätzung von Unsicherheiten umgesetzt und für die Detektion von Modell- und Datenunsicherheit eingesetzt werden. Die Ergebnisse konnten weiterhin schrittweise auf die Detektion von Straßenschäden auf Oberflächenbildern übertragen werden. Hierbei zeigten sich vor allem für die Modellunsicherheit vielversprechende Ergebnisse, während die Schätzung der Datenunsicherheit noch nicht sinnvoll für das Projekt umgesetzt werden kann. Das eigentlich verfolgte Ziel sollte die Identifikation von problematischen Objekten/Bildern sein, die zu Widersprüchen in den Daten führen und deren Labeling daher ungeeignet ist. Die Ergebnisse, die mit Patch-basierten Ansätzen erreicht werden konnten, waren jedoch in Hinblick auf die Verwertbarkeit nicht zielführend. Jedoch spielt die Datenunsicherheit im Rahmen des Projektes ASFaLT nur eine untergeordnete Rolle und wird für die spätere Anwendung der Netzwerke nicht benötigt.

Für das Projekt hat sich die Schätzung von Unsicherheiten über Monte-Carlo-Dropout und alternativ über Ensemble-Ansätze als sinnvolle Methoden für die Schätzung der Unsicherheit herausgestellt. In Untersuchungen zeigte sich dabei, dass bei Monte-Carlo-Dropout deutlich mehr Durchläufe generiert werden müssen, um stabile Aussagen zur Sicherheit zu erhalten. In Hinblick auf die Laufzeit, welche linear mit der Anzahl der Durchläufe skaliert, wird daher im Projekt ASFaLT ein Ensemble-Ansatz für die Schätzung von Unsicherheiten auf Segmentierungsansätzen favorisiert.

5.6 *Modellunsicherheiten für Segmentierungsansätze*

Die in den vorherigen Teilkapiteln vorgestellten Verfahren zur Bestimmung von Unsicherheiten basieren auf der Grundlage von Patch-basierten Netzwerken. Das bedeutet, dass die Netzwerke jeweils nur auf einem kleinen Ausschnitt des Gesamtbildes angewendet werden, um für diesen Bildbereich genau eine Klassenausgabe zu generieren. Um für das gesamte Eingabebild eine räumlich detailliertere Ausgabe erzielen zu können, wird das Netzwerk mehrmals und immer räumlich versetzt auf dem kompletten Bild angewendet.

Diese Art der Klassifikation führt zu Klassifikationen, deren Ausgaben eine gewisse Unschärfe aufweisen und in der Regel auch keine pixelgenaue Klassifikation erlauben.

Auch bei der Schätzung der Unsicherheit zeigt sich eine große Unschärfe, wie in Abbildung 31 zu sehen ist.

Um die Modellunsicherheit abschätzen zu können, benötigt man in der Regel mehrere Samples aus der A-posteriorio-Verteilung der Problemstellung. So erfolgt beispielsweise bei den optimierungsbasierten Ansätzen mit Monte-Carlo Dropout [Gal 2016b] während der Anwendung ein Sampling, wodurch mehrere Parameterkonfigurationen des Netzwerks auf das jeweilige Oberflächenbild angewendet werden.

Durch dieses Vorgehen erfolgt eine lokale Abschätzung der A-posteriorio-Verteilung, welche Rückschlüsse auf die Unsicherheit der Schätzung ermöglicht. Als nachteilig erweist sich, dass in der Regel viele Parameterkonfigurationen des Netzwerks angewendet werden müssen, um eine robuste Schätzung der Unsicherheit erhalten zu können, was die Laufzeit der Schadstellendetektion deutlich erhöht.

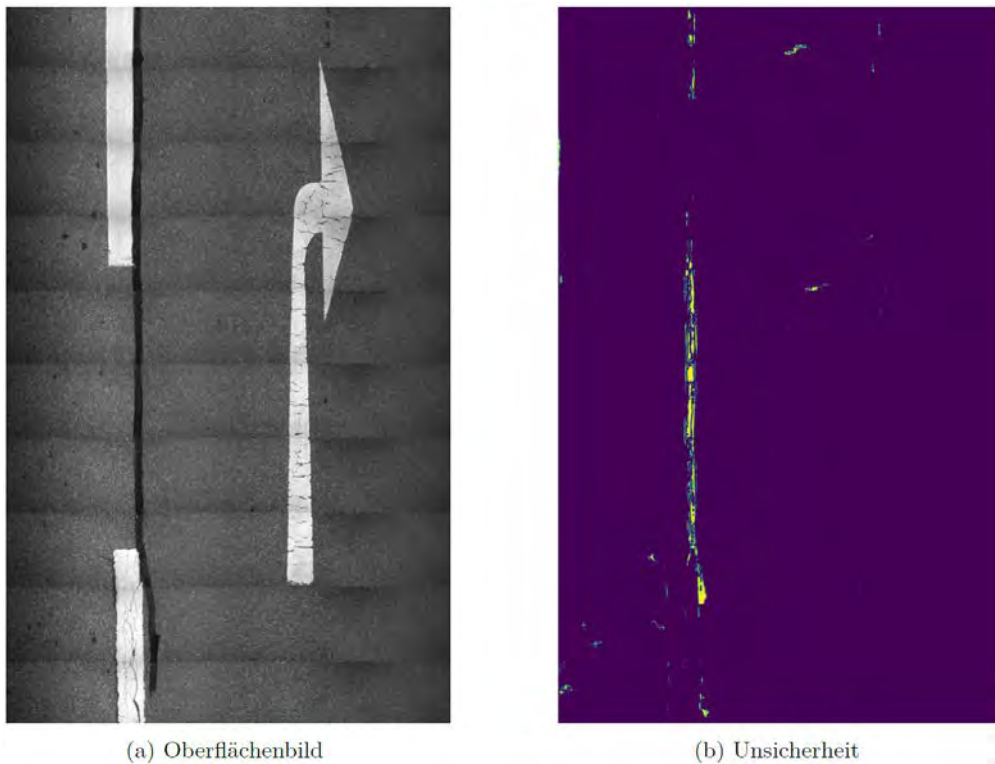


Abbildung 32: Über die Schätzung der Unsicherheit der Klassifikation des Oberflächenbildes (a) mit Segmentierungsnetzwerken im Ensemble-Verbund ergeben sich gut lokalisierbare Unsicherheiten (b). Unsicher ist beispielsweise die Klassifikation des verfüllten Risses, dessen Klassifikation im mittleren Bereich zwischen Naht-riss und verfülltem Riss schwankt. (lila - geringe Unsicherheit, gelb - hohe Unsicherheit)

Die Abschätzung der Unsicherheiten mit Ensembles als Vertreter aus der Gruppe der Nicht-bayesschen Ansätze [Lakshminarayanan 2017] hat sich in Praxistests ebenfalls als sehr robustes Verfahren erwiesen, welches in der Anwendungsdomäne mit weniger Samples eine aussagekräftige Schätzung erzielen kann. So hat sich in Tests gezeigt, dass bereits mit drei divers trainierten Netzwerken eine Schätzung möglich ist, welche zu plausiblen Ergebnissen führt. Zwar ergibt sich dadurch immer noch eine wesentliche Verlangsamung

der Klassifikation (Faktor 3), allerdings hält sich diese in direktem Vergleich zu Monte-Carlo-Verfahren in Grenzen und kann durch Heuristiken, welche in Abschnitt 5.8 vorgestellt werden, noch weiter reduzieren.

Im direkten Vergleich zu den Unsicherheiten, die mit den Patch-basierten Netzwerken erzeugt wurden, zeigt sich, dass die Unsicherheiten eine deutlich höhere räumliche Auflösung besitzen und auch eine *schärfere* Verortung möglich ist (Abbildung 32).

5.6.1 Unsicherheiten an Klassengrenzen

An Klassengrenzen, also beispielsweise an dem Übergangsbereich zwischen Riss und schadenfreiem Straßenbelag, ergeben sich in der Regel Bereiche mit größerer Unsicherheit. Diese Unsicherheiten sind eigentlich der Datenunsicherheit zuzuordnen, da sie, bedingt durch die unscharfe Begrenzung der Klasse, aufgrund von nicht eindeutigem Labeling der Übergangsbereiche entstehen. Allerdings lässt sich in diesem Fall kaum eine scharfe Grenze zwischen beiden Unsicherheitstypen ziehen, da die Unsicherheit beispielsweise auch entstehen könnte, weil die Struktur der Asphaltoberfläche von den Trainingsdaten abweicht und somit der Übergangsbereich auch nicht mehr vollständig durch die Trainingsdaten repräsentiert ist. Folglich kommt es an Übergangsbereichen zwischen unterschiedlichen Klassen (auch wenn diese sehr sicher geschätzt werden können) in vielen Fällen zu einem Ausschlag bei der Modellunsicherheit wie in Abbildung 33(b) zu sehen ist.

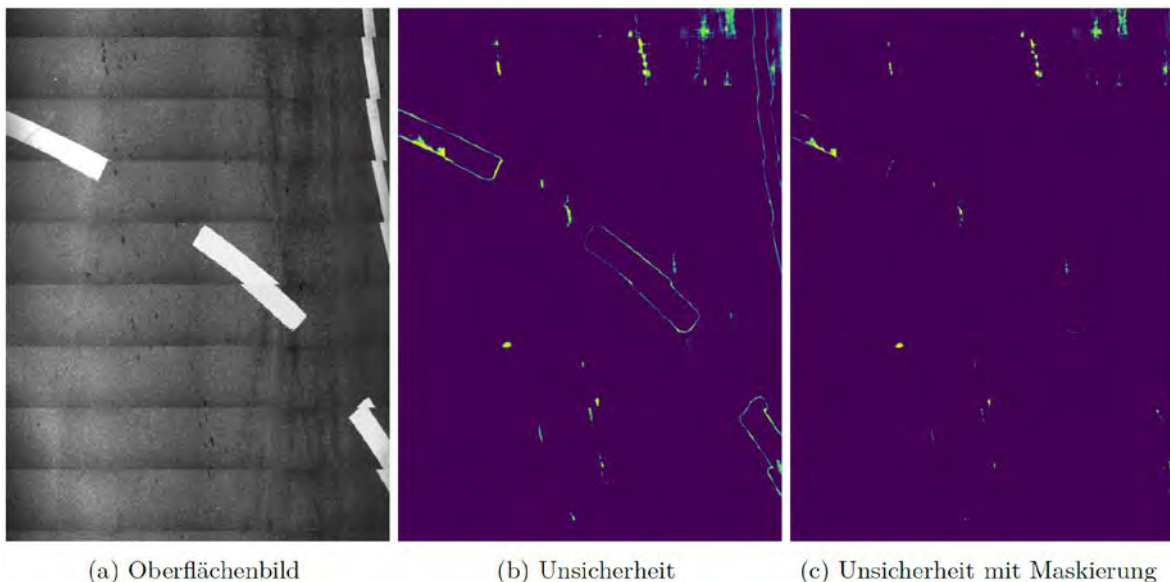


Abbildung 33: Für das Oberflächenbild (a) ergeben sich insbesondere an den Klassenübergängen, gut zu sehen an den Markierungen, Unsicherheiten (b). Durch Maskierung der Unsicherheiten an den Klassengrenzen können diese Unsicherheiten gut unterdrückt werden (c). (lila - geringe Unsicherheit, gelb - hohe Unsicherheit)

Diese Übergänge würden insbesondere bei Bildern mit vielen Schadstellen oder Objekten automatisch zu einer hohen Bewertung mit Unsicherheit führen. Um diesen Umstand Rechnung zu tragen, werden die Unsicherheiten an Klassengrenzen explizit maskiert. Dazu wird über jeder Modelldetektion das Kantenbild erstellt und die detektierten Übergangsbereiche eine Objektübergangsmaske hinzugefügt. Durch multiplikative Verrechnung mit der Unsicherheitskarte können die Unsicherheiten an Klassenübergängen damit effektiv beseitigt werden (Abbildung 33 (c)).

5.6.2 Bewertung der Modellunsicherheit

Nachdem für das Bild eines Streckenabschnitts eine Unsicherheitskarte berechnet wurde, muss diese automatisiert bewertet werden. Hierbei erscheint zuerst eine Normierung der Unsicherheit auf eine bestimmte Region sinnvoll, um Unsicherheiten von der eigentlichen Bildauflösung und damit von der Abhängigkeit des länderspezifischen Aufnahme Standards zu entkoppeln. Daher wird im Projekt die Unsicherheit jeweils für 1 qm große, quadratische Fläche betrachtet.

Bei der genaueren Betrachtung der Unsicherheitskarten tritt ein Problem in den Vordergrund, welches sich aus der stark unterschiedlichen Größe der zu detektierten Objekte und Schadensstellen ergibt. Wenn beispielsweise eine geringe Unsicherheit bei einem großflächigen Objekt auftritt, würde sich dies, bei linearer Addition aller Unsicherheitswerte σ für jeden Pixel in der Bezugsregion, stärker auswirken als eine große Unsicherheit in einer kleinen lokalen Region. Die Unsicherheiten werden daher nichtlinear addiert, um großen Unsicherheiten, die einer manuellen Inspektion bedürfen, ein größeres Gewicht zukommen zu lassen. Die Nichtlinearität für die Bestimmung der Unsicherheit wurde empirisch bestimmt und ergibt sich aus folgender Formel:

$$\sigma_n = \frac{10^\sigma - 1}{10 - 1}$$

Die somit berechnete Unsicherheit pro Meterzelle wird anschließend als Kriterium für die Selektion von Bildern verwendet, deren Klassifikation noch einmal manuell überprüft werden sollten.

5.7 Selektion von Bildern mit großer Unsicherheit für das Labeling

Die Ermittlung von Modellunsicherheiten wurde in der ersten Jahreshälfte 2020 bereits eingesetzt, um aus ungelabelten Oberflächenbildern diejenigen auszuwählen, deren Labeling sich für die Verbesserung des Klassifikators besonders lohnen würde. Dazu wurden 978

Bilder nach deutschem Standard (10-Meter-Bilder) und 324 Bilder nach österreichischem Standard (4x3-Meter-Bilder) auf ihre Unsicherheit analysiert.

Dabei konnten vor allem unbekannte Oberflächenstrukturen ausfindig gemacht werden, welche in dem bisherigen Datensatz noch nicht repräsentiert waren und daher zu einer unterschiedlichen Klassifikation durch die Netzwerke geführt haben (Abbildung 34). Die Unsicherheiten wurden den Projektpartnern zur Verfügung gestellt, welche die Unsicherheiten bei der Auswahl der Bilder für das nächsten Labeling berücksichtigt haben. Somit konnte der Datensatz für das Training des Klassifikators durch den Einsatz von Unsicherheiten gezielt erweitert werden.

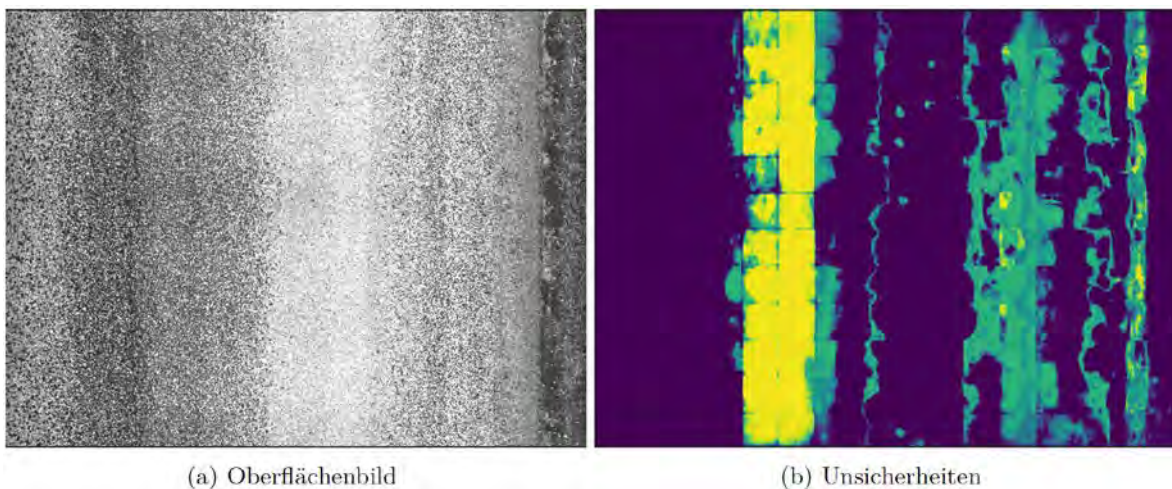


Abbildung 34: Die Schätzung der Unsicherheiten führt bei Oberflächenstrukturen, welche noch nicht von den Trainingsdaten abgedeckt werden (a) zu großflächigen Ausschlägen in der Unsicherheitskarte (b). (lila - geringe Unsicherheit, gelb - hohe Unsicherheit)

5.8 Einsatz von Unsicherheiten für die Bewertung von Messfahrten

Für die Bewertung von Messfahrten spielt der Einsatz von Unsicherheiten eine große Rolle, um beurteilen zu können, ob eine Messfahrt oder Teile einer Messfahrt einer zusätzlichen manuellen Analyse bedürfen. Hierbei gilt es vor allem zur berücksichtigen, dass die zusätzliche Bestimmung der Unsicherheiten mit einer Erhöhung der Laufzeit einhergeht. Daher werden aktuell zwei Strategien verfolgt, um den Einsatz von Unsicherheiten bei der Analyse von Messfahrten zu beschleunigen.

Der erste Ansatz basiert auf der Tatsache, dass Unsicherheiten fast ausschließlich bei der Klassifikation von unterschiedlichen Schadens-/Objekttypen auftreten. Es ist aber selten der Fall, dass ein Detektor einen Abschnitt komplett als schadenfrei klassifiziert, während ein anderer Detektor auf dem Abschnitt mehrere Schadens- oder Objekttypen findet. Ba-

sierend auf dieser Annahme könnte die Berechnung von Unsicherheiten nur für die Abschnitte erfolgen, auf denen auch tatsächlich unterschiedliche Schadens- und Objektklassen detektiert wurden (Markierungen ausgenommen).

Ein zweiter Ansatz besteht aus der Berechnung der Unsicherheiten auf einer kleineren Bildskalierung. Als nachteilig erweist sich bei diesem Ansatz jedoch, dass aufgrund der unterschiedlichen Auflösung zusätzliche Detektoren trainiert werden müssen und auch die Erfassung von feinen Strukturen nur noch eingeschränkt möglich ist.

Eine Umsetzung der Beschleunigungstechniken konnte im Rahmen des Projekts ASFaLT nicht mehr erfolgen und könnte in nachfolgenden Forschungsprojekten intensiver betrachtet werden.

6 Unterscheidung unterschiedlicher Straßenoberflächen

Nachdem in den vorhergehenden Abschnitten die Detektionsleistung auf den jeweiligen Oberflächenarten getrennt voneinander betrachtet wurde, muss für die praktische Anwendung auch der korrekte Klassifikator für die jeweilige Oberfläche ausgewählt werden. Dazu ist es erforderlich zu detektieren, welche Bauweise der aktuellen Aufnahme zugrunde liegt. Diese Problemstellung lässt sich, für großflächige Änderungen der Bauweise, sehr einfach durchführen. Die regelmäßigen Fugen, welche bei der Betonbauweise entstehen, lassen sich recht leicht detektieren, sodass eine Unterscheidung über einen einfachen Vorklassifikator durchgeführt werden kann.

Im Rahmen des Projekts sollte jedoch auch untersucht werden, ob sich die Bauweisen auch ohne dieses prominente Merkmal und nur anhand der Oberflächenstruktur unterscheiden lassen. Dies würde die Erkennung von kleinen Bereichen in anderer Bauweise ermöglichen. Darüber hinaus könnte auf diese Weise auch ein System trainiert werden, welches zusätzlich die Unterscheidung unterschiedlicher Ausprägungsarten der Bauweisen ermöglicht. Wenn ein Detektor die vorliegende Oberflächenstruktur jeweils einem bestimmten Merkmalsvektor zuordnet, könnten bereits leichte Veränderungen der Struktur detektiert werden. Dies würde die Detektion von Flickstellen, welche sich typischerweise von der umgebenden Bebauung in der Struktur unterscheiden, ermöglichen und könnte als zusätzliches Merkmal in die Schadstellendetektoren für Beton und Asphalt einfließen. Der Fokus dieses Abschnitts liegt daher auf der Unterscheidung von Asphalt und Beton auf Grundlage der Oberflächenstruktur (Bildausschnitte ohne weitere Objekte wie beispielsweise Fugen) und der Überführung des Problems in die Schätzung von Merkmalsvektoren für Bildausschnitte von Oberflächenbildern.

Grundlage für die in diesem Abschnitt vorgestellten Ergebnisse bildet die im Rahmen des Projekts durchgeführte Masterarbeit von Friedrich Schmidt [Schmidt, 2021].

6.1 *Datengrundlage*

Basis für die Untersuchungen bildet der für die Schadstellenerkennung aufgebaute Datensatz mit dem Stand aus dem Juni 2020 (390 Bilder Asphalt, 190 Bilder Beton). Um die Diversität des Datensatzes zu erhöhen, wurde, unter Zuhilfenahme des IT-ZEB-Servers, die Datenbasis erweitert. Dazu wurden über die ZEB-Zustandsnoten Straßen ausgewählt, welche die besten 100-Meter-Abschnitte aufweisen. Das erste Bild des besten Abschnitts einer Straße wurde anschließend in den Datensatz aufgenommen. Für die Oberflächenart Asphalt wurden auf diese Weise weitere 89 Bilder in den Datensatz aufgenommen und für die Oberflächenart Beton weitere 127 Bilder.

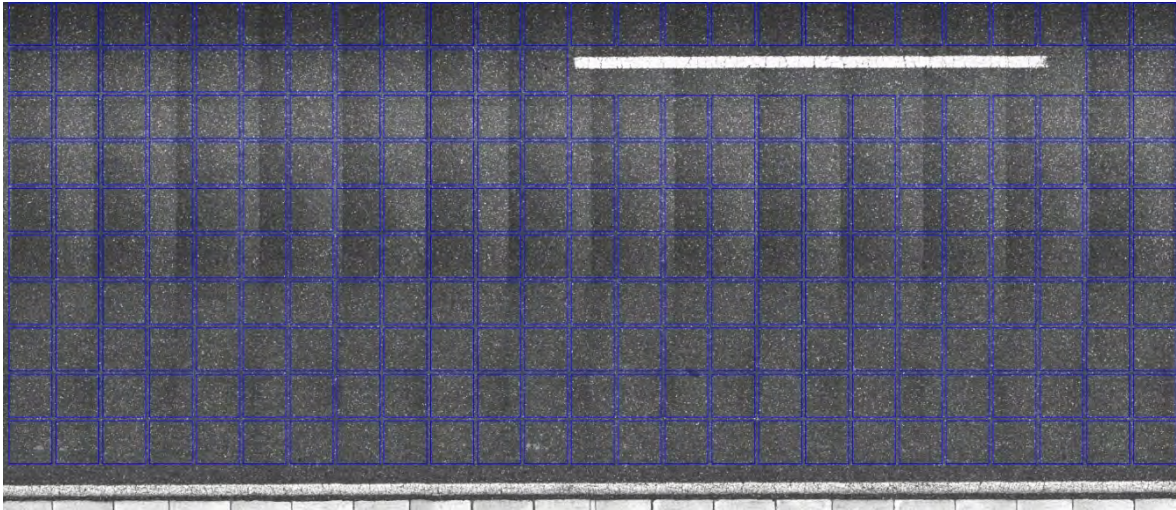


Abbildung 35: Beispielhafte Darstellung der aus den Oberflächenbildern extrahierten Patches (blau) [Schmidt, 2021].

Aus den Bildern werden jeweils Patches einer festen Größe extrahiert, wobei nur Bereiche berücksichtigt werden, die den Normalzustand der Straße zeigen (Abbildung 35). Die Patches enthalten somit keine Objekt- oder Schadenstypen, deren Auftreten eine Zuordnung zu einer bestimmten Oberflächenart zulässt.

6.2 *Unterscheidung von Beton und Asphalt*

Zunächst wurde untersucht, wie gut, nur über Bildpatches der jeweiligen Oberflächenart, eine Unterscheidung von Beton und Asphalt möglich ist. Diese Problemstellung kann supervised, also mit bekannter Zuordnung der Zielklassen, trainiert werden, da für die Bilddaten bekannt ist, ob diese auf einer Asphalt- oder Betonstraße aufgenommen wurden.

Folglich erfolgt das Training analog zu den Trainingsläufen der Schadstellendetektoren. Der wesentliche Unterschied besteht in der Netzwerkstruktur. Die Netzwerke werden als reine Patch-basierte Klassifikatoren trainiert. Entsprechend findet für diese Problemstellung nur der *Encoderblock* in Form eines Resnets mit 18 Schichten Verwendung.

Das Training erfolgte mit den üblichen Augmentierungstechniken (Helligkeit, Kontrast, Rauschen, Rotation) und führte auf den Trainingsdaten bereits nach wenigen Epochen zu einer sehr guten Anpassung an die Daten. Auf den Validierungsdaten zeigte sich jedoch eine deutlich schlechtere Klassifikationsleistung und auch ein sehr unsteter und sprunghafter Trainingsverlauf.

6.2.1 Normalisierung von Bildpatches

Ein wahrscheinlicher Grund für dieses Verhalten ist in Abbildung 36 zu sehen.

In den Daten scheint es einen sehr starken Zusammenhang zwischen Oberflächenart und Bildhelligkeit zu geben. Wobei Betonstraßen tendenziell im Bild heller abgebildet werden als Asphaltstraßen.

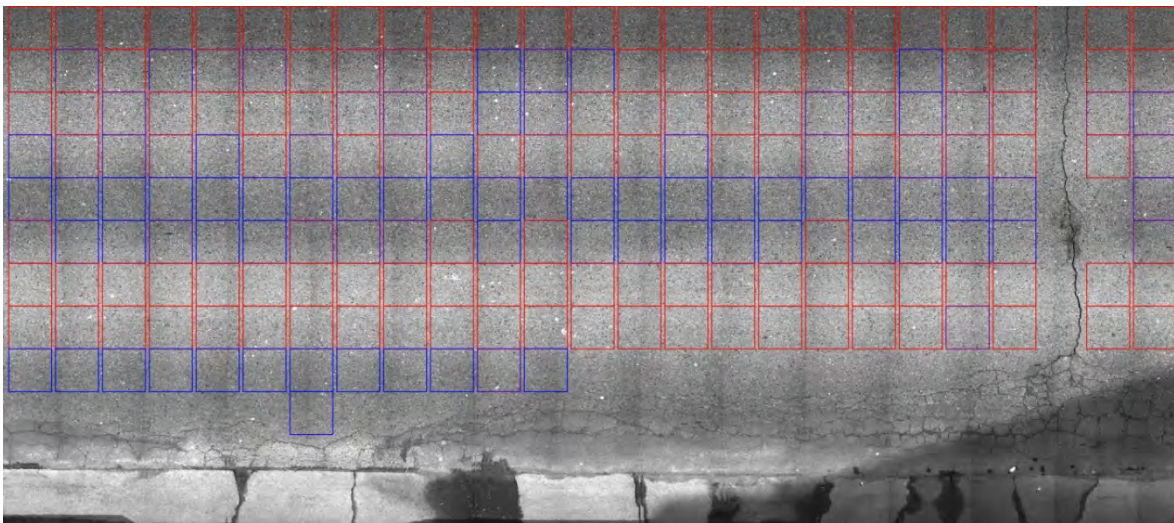


Abbildung 36: Klassifikation von Bildpatches eines Oberflächenbildes. Rote Patches werden sehr sicher der Oberflächenklasse Beton zugeordnet - blaue Patches der Klasse Asphalt. Farbkodierungen dazwischen weisen auf unsichere Schätzungen hin. Obwohl das Oberflächenbild eine Asphaltoberfläche darstellt, erfolgt scheinbar helligkeitsabhängig eine Zuordnung zur Oberflächenart Beton. [Schmidt, 2121]

Dies führt selbst innerhalb eines Bildes zu einem häufigen Wechsel der vorhergesagten Oberflächenklasse, sobald ausgeprägte Helligkeitsunterschiede innerhalb des Bildes auftreten.

Um diesen Effekt entgegenzuwirken, wurden die Bildpatches in der Vorverarbeitung helligkeitsnormiert, sodass das Netzwerk die mittlere Bildhelligkeit nicht mehr als Klassifikationsmerkmal verwenden kann und gezwungen ist, die Struktur stärker zu berücksichtigen.

6.2.2 Ergebnisse

Im Rahmen des Projekts wurden auch weitere Faktoren systematisch untersucht. Neben Modifikationen der Trainingsparameter und Algorithmen wurde beispielsweise unterschiedliche Patchgrößen und Patchextraktionsalgorithmen analysiert. Mit den vorgenommenen Anpassungen konnte auf dem Validierungsdatensatz ein F1-Wert von 94% erzielt werden, was durchaus als ein sehr guter Wert angesehen werden kann. Außerdem lässt sich dieser Wert für die Klassifikation ganzer Oberflächenbilder weiter steigern, da diese auf einer Vielzahl von Patches bestehen und somit eine Mehrheitsentscheidung getroffen werden kann, welche die Güte weiter steigert. Allerdings ist zu berücksichtigen, dass auch mit allen Optimierungen des Trainings und unter Nutzung des vergrößerten Datensatzes eine große Streuung der Validierungsergebnisse zu beobachten ist. Da diese Streuung von bis zu 15% auch in einem sehr weit fortgeschrittenem Trainingsprozess auftritt, ist davon auszugehen, dass die Trainingsdaten noch nicht repräsentativ genug sind, um in jedem Fall auf den Validierungsdaten gute Ergebnisse zu erzielen. Naheliegenderweise zeigt sich auf den Testdaten ein ähnliches Verhalten. Auf diesen Daten wird ein F1-Wert von maximal 90,6% erreicht. Allerdings führt die Auswahl des besten Modells auf den Validierungsdaten nicht zwingend zu einem guten Ergebnis auf den Testdaten, so dass die F1-Werte auf den Testdaten zwischen 80% und 90% schwanken können. Um diese Effekte weiter zu untersuchen, müssten die Trainingsdaten daher noch einmal deutlich erweitert werden. Grundsätzlich ist jedoch zu beobachten, dass die unterschiedlichen Oberflächenarten sehr große Ähnlichkeiten aufweisen können (Abbildung 37).

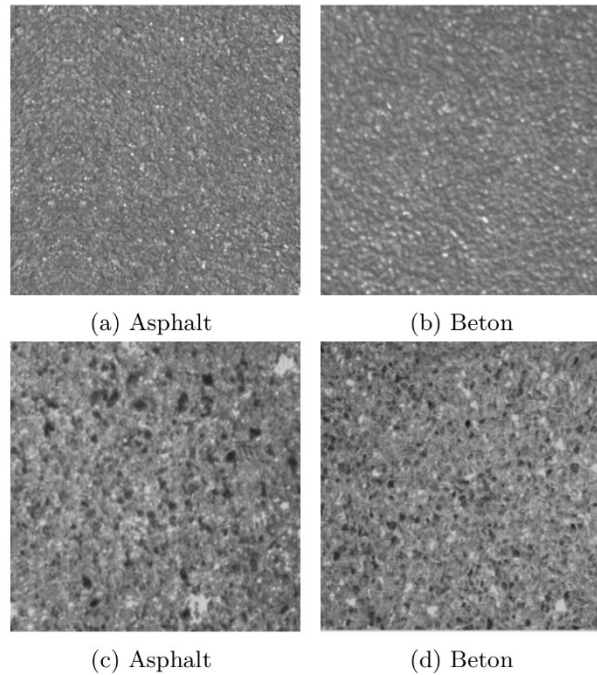


Abbildung 37: Beispielhafte Bildpatches der unterschiedlichen Oberflächenklassen, welche sehr große Ähnlichkeiten aufweisen. [Schmidt, 2021]

6.3 Unterscheidung unterschiedlicher Ausprägungen von Oberflächenarten

Insbesondere für die Detektion von Flickstellen kann die Bestimmung eines Oberflächeneigenschaftsvektors für Fahrbahnoberflächen nützlich sein. Dabei geht es nicht darum, die Bildausschnitte einer festen Klasse zuzuordnen. Vielmehr soll es über Distanzen zwischen den Eigenschaftsvektoren möglich sein, seine Ähnlichkeiten zu detektieren, um beispielsweise Flickstellen über die Änderung der Oberflächenstruktur erkennen zu können. Dies ist auch im weiteren Prozess der Datenverarbeitung des Projekts hilfreich, da über die Schadensstellendetektoren für Beton und Asphalt nur die Umrandung von Flickstellen detektiert wird. Die flächenhafte Zuordnung zu normaler Belagstruktur und zu Flickstellen kann unter Zuhilfenahme des Oberflächeneigenschaftsvektors erfolgen.

6.3.1 Datengrundlage

Grundlage für die Untersuchung bildet erneut der unter 6.1 beschriebene Bilddatensatz. Für die Unterscheidung unterschiedlicher Ausprägungen von Oberflächenarten wurden für den Datensatz zusätzliche Unterteilungen vorgenommen:

Two

Die erste Möglichkeit besteht darin, alle Patches nach dem Typ ihres groben Straßenbelages einzuteilen, womit es die beiden Label Asphalt und Beton gibt. Der Datensatz würde das Netzwerk also darauf trainieren, die Oberflächenstruktur von Asphalt und Beton zu unterscheiden. Wichtig ist hier zu beachten, dass Betonstraßen teilweise mit Flickstellen aus Asphalt repariert werden und diese Patches damit unter die Kategorie Asphalt fallen. Bei dieser Einteilung umfasst der Trainingsdatensatz 87012 Patches und zwei verschiedene Label.

Groups

Die zweite Möglichkeit ist die Einteilung der Patches in die jeweiligen Lokalisationsgruppen ihres Rohbildes, um die Oberflächenstrukturen der jeweiligen Gruppen zu lernen. Bei dieser Variante der Labelzuordnung werden alle Patches von Flickstellen ignoriert und verworfen, da diese Fehlinformationen über die eigentliche Struktur der Straßen innerhalb der Lokalisationsgruppe ins Training einbringen würden. Daher ist dieser Datensatz mit 86245 Trainingspatches und 158 Labeln etwas kleiner.

Groups+

Um keine Patches und damit die Informationen über Flickstellen zu verlieren, erhalten die Flickstellen bei dieser Variante jeweils eine eigne Lokalisationsgruppe. Damit erreicht der Trainingsdatensatz wieder seine volle Größe von 87012 Patches und erweitert sich um 25 Lokalisationsgruppen auf damit insgesamt 183 verschiedene Label. Für alle unsupervised Verfahren ist diese Einteilung identisch zur zwei-Klassen-Einteilung.

Img

Die letzte verwendete Labelzuordnung ist die Einteilung der Patches nach Rohbild. Bei diesem Datensatz würde also versucht werden die Oberflächenstruktur von jedem einzelnen Rohbild zu lernen. Auch hier wäre die Verwendung der Flickstellen wieder kontraproduktiv. Daher umfasst der Trainingsdatensatz bei dieser Variante wieder nur 86245 Patches aber 548 Label.

6.3.2 Training

Auch wenn die Klassifikation, also die Zuordnung zu festen IDs, nicht das Ziel dieser Problemstellung ist, kann das Training über klassisches *supervised*-Learning auf den unter 6.3.1 aufgeführten Datensätzen durchgeführt werden. Um den für die Problemstellung erwünschten Merkmalsvektor der Oberfläche zu erhalten, wird von dem verwendeten Netzwerk anschließend die Klassifikationsschicht entfernt. Somit übernimmt die letzte Faltungsschicht des Netzwerks die Abbildung in den Merkmalsraum.

Als Alternativen wurden Verfahren aus der Gruppe des *Deep-Clustering* und des *Similarity-Learnings* untersucht. Deep-Clustering-Verfahren gruppieren ähnliche Daten auf Grundlage von Distanzmetriken und können damit völlig ohne Labelinformationen trainiert werden. Die Verfahren aus der Gruppe des Similarity-Learnings versuchen die, von einem Netzwerk gelernten, Distanzen in einem internen Merkmalsraum für Daten der gleichen Klasse zu reduzieren und Distanzen zwischen Daten von unterschiedlichen Klassen zu erhöhen.

6.3.3 Ergebnisse

Im Rahmen des Projekts wurden eine Reihe unterschiedlicher Untersuchungen mit klassischem Supervised-Learning, Deep-Clustering und Similarity-Learning durchgeführt. Da verschiedene Kombinationen jedoch keine guten Resultate erzielen konnten, wird in dieser Zusammenfassung nur auf den Vergleich zwischen dem Ansatz mit klassischem Supervised-Learning und einem Vertreter des Similarity-Learnings eingegangen, da diese die vielversprechendsten und repräsentativsten Ergebnisse erzielen konnten.

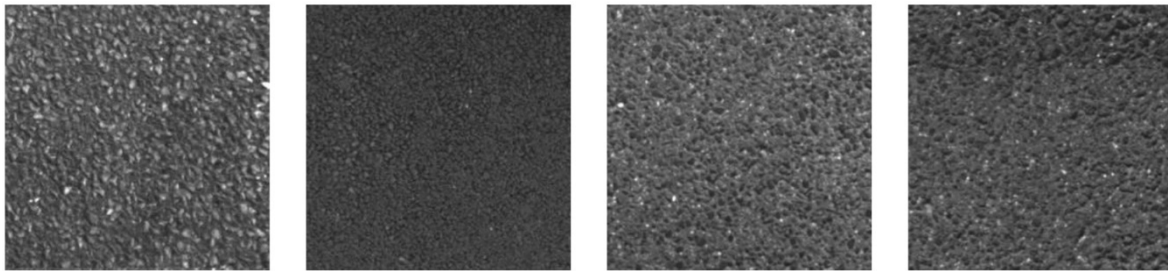
In den Untersuchungen wurden beide Ansätze zunächst auf den in 6.3.1 vorgestellten Datensätzen trainiert. Anschließend wurden die Testdaten mit den Netzwerken in den jeweiligen internen Merkmalsraum projiziert. Um bewerten zu können, wie gut der von den Netzwerken gelernte Merkmalsraum für die Unterscheidung von Normalbelag und Flickstellen geeignet ist, wurden die in den internen Merkmalsraum transformierten Daten über den k-Means Algorithmus geclustert. Korrekt dem jeweiligen Cluster zugeordnete Patches werden als richtig klassifiziert gewertet, anderenfalls als falsch. Diese Bewertung lässt sich wieder in den F1-Wert überführen, sodass sich die in Tabelle 12 gegebenen Ergebnisse erzielen lassen.

Tabelle 12: F1-Werte für die Flickstellenunterscheidung auf Grundlage von unterschiedlichen Detektionsalgorithmen.

| Modellauswahl | Flickstellen | | | |
|----------------------|---------------------|--------|---------|-------|
| | Two | Groups | Groups+ | Img |
| <i>Supervised</i> | 82,0% | 80,1% | 76,1% | 74,5% |
| <i>Similarity L.</i> | 82,2% | 78,5% | 77,0% | 78,5% |

Es zeigt sich, dass sich, sowohl zwischen den einzelnen Datensätzen als auch zwischen den unterschiedlichen Trainingsansätzen, kaum nennenswerte Unterschiede feststellen

lassen. Insbesondere in Anbetracht des kleinen Datensatzes lassen sich somit kaum Unterschiede in der Eignung der unterschiedlichen Methoden feststellen. Es bleibt jedoch festzuhalten, dass die grundsätzliche Unterscheidbarkeit in den meisten Fällen durchaus gegeben ist und die internen Merkmalsräume das Potenzial haben, zwischen unterschiedlichen Oberflächenausprägungen zu unterscheiden.



(a) Oberflächenbild 1 -
Originale Straßenober-
fläche

(b) Oberflächenbild 1 -
Flickstelle

(c) Oberflächenbild 2 -
Originale Straßenober-
fläche

(d) Oberflächenbild 2 -
Flickstelle

Abbildung 38: Vergleich der Oberflächenstrukturen von Flickstellen zum originalen Straßenbelag Die Bilder (a) und (c) sind Bilder des originalen Straßenbelages wohingegen die Bilder (b) und (d) Patches aus Flickstellen auf diesen Straßen entnommen sind. [Schmidt, 2021]

Für eine qualitative Betrachtung soll abschließend das Bild mit der besten Trennbarkeit zwischen Oberfläche und Flickstelle und das Bild mit der schlechtesten Trennbarkeit betrachtet werden. In Abbildung 38 ist zu erkennen, dass die Änderungen der Oberflächenstruktur zwischen normaler Oberfläche und der Flickstelle auf Oberflächenbild 1 deutlich ausgeprägter ist als auf dem Oberflächenbild 2. Folglich ist die Trennung für die Netzwerke auf dem ersten Bild einfach möglich, während bei Oberflächenbild 2 keine gute Trennbarkeit erreicht werden kann. Dennoch scheint sich der Merkmalsvektor, in der Breite der Anwendung, gut als zusätzlicher Input für die Detektion und die Bestimmung der Fläche von Flickstellen nutzen zu lassen.

7 Abbildung der Automatisierten Detektion auf die länderspezifischen Bewertungen

Für die Nutzbarkeit der im ASFaLT-Projekt entwickelten Software spielt die Abbildung der Detektionsergebnisse in die länderspezifischen Dateiformate eine wichtige Rolle, da nur so die unmittelbare Verwendung im Rahmen der klassischen Zustandserfassung sichergestellt werden kann. Ausgehend von den Detektionen des Schadens- und Objektklassifikators müssen die Klassen zuerst in die jeweils länderspezifischen Schadensklassen abgebildet werden. Anschließend kann eine Speicherung in dem länderspezifischen

Dateiformat vorgenommen werden. Die Umsetzung beider Schritte wird in den folgenden Abschnitten beleuchtet.

7.1 *Abbildung von Detektionsklassen in Schadensklassen*

Der im Projekt erarbeitete universelle Katalog von Labelklassen bildet die Grundlage für das hochgenaue Labeling der Trainingsdaten und definiert somit die Zielklassen für den Schadens- und Objektklassendetektor. Die festgelegten Klassen sollten ein möglichst breites Spektrum an Schadens- und Objekttypen umfassen, mit deren Auftreten auf und neben der Straßenoberfläche zu rechnen ist. Außerdem sollen Klassen berücksichtigt werden, denen in Hinblick auf zukünftige Bewertungsstandards Relevanz zukommen könnte. Es wurde jedoch ebenso darauf geachtet, dass sich die Labelklassen auf die derzeit gültigen Schadensklassen für die Zustandsbewertung in den D-A-CH-Ländern abbilden lassen (siehe 3.1). Die erarbeitete Projektionstabelle liefert daher bereits einen Leitfaden für die Überführung der detektierten Labelklassen in die länderspezifischen Schadensklassen. Bei einigen Schadensarten müssen die Detektionen jedoch noch weiterverarbeitet werden, um die für die Zustandsbewertung notwendigen Informationen zu extrahieren. Dies betrifft neben der Bestimmung von Längen- und Flächenmaßen auch die Unterscheidung zwischen Einzel- und Netzfalten, welche nicht durch separate Labelklassen abgebildet werden.

7.2 *Nachverarbeitung der Detektionsergebnisse*

Die von den Schadstellendetektoren produzierten Segmentierungskarten sind in der Regel mit einem gewissen „Rauschen“ behaftet, welches bei einer menschlichen Bewertung nicht vorliegen würde. So kommt es beispielsweise zur Detektion von sehr kleinen Risskandidaten oder auch zu schwankenden Klassenbewertungen entlang von Objektgrenzen (siehe Abbildung 39 (a), (b)). So kommt es an Objektgrenzen beispielsweise häufig zu einer schwankenden Detektion zwischen den 3 optisch sehr ähnlichen Klassen Naht, EFLI und sanierter Riss.

Um die Detektion auch ohne die Zuhilfenahme weiterer Trainingsdaten zu verbessern, wurde eine Schadstellenheuristik entwickelt, welche häufige Fehlerursachen regelbasiert behebt. Die Detektion lässt sich dabei in 4 aufeinander aufbauende Schritte einteilen:

1. Kontrasterhöhung

Um Mehrdeutigkeiten zwischen den Klassen Naht, EFLI und sanierter Riss aufzulösen, wird zunächst eine Analyse der Form der Objekte auf der kombinierten Detektionskarte dieser

3 Schadenstypen vorgenommen. Dazu wird für jede erkannte Objektinstanz auf der kombinierten Detektionskarte die Hough-Linientransformation berechnet. Wenn ein Objekt erkannt wurde, welches sehr geradlinig ist, wird die Netzwerkausgabe für die Klassen EFLI und Naht erhöht und für die Klasse versiegelter Riss erniedrigt. Inverses Vorgehen wird angewendet, wenn die Detektion keine geradlinige Form aufweist (Abbildung 39 (c)).

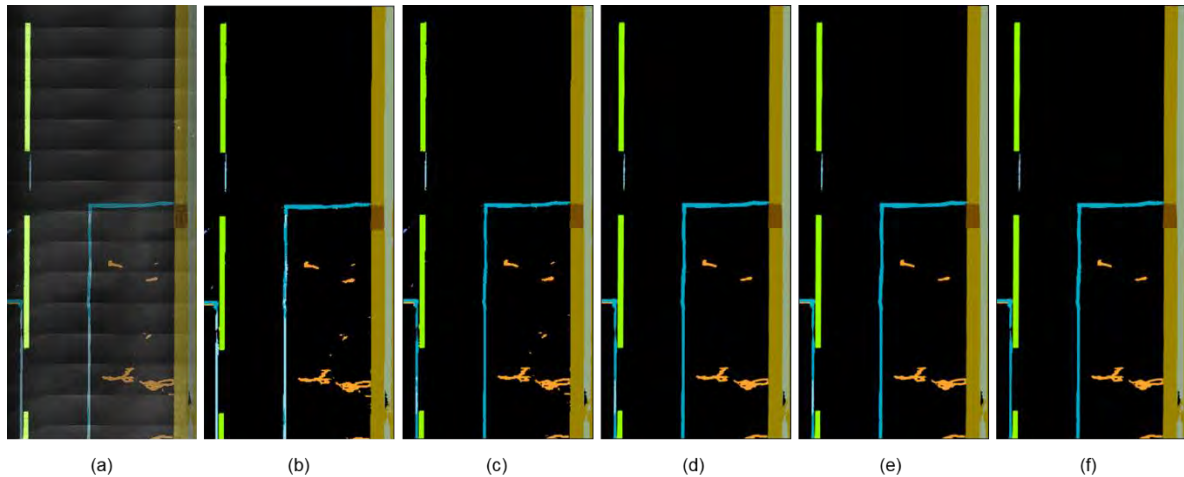


Abbildung 39: Nachverarbeitung von Segmentierungsergebnissen. (a) Unverarbeitete Ausgabe des Detektors (als Überlagerung mit Oberflächenbild). Entlang der Objektgrenze der EFLI gibt es schwankende Klassenentscheidungen; Detektion von sehr feinen und kleinen Rissen. (b) Unverarbeitete Ausgabe des Detektors. (c) „Kontrasterhöhung“ durch Filterung von Klassen anhand von Formnebenbedingungen (d) Filterung von Klassen aufgrund von Größe/Länge im „Logit Raum“ (e) Filterung von Klassen nach Maximums-Auswahl (f) Mehrheitsentscheid

2. Klassenfilterung („Logit-Raum“)

Im zweiten Schritt werden minimale Detektionen unterdrückt. Dazu werden für alle Objektdetektionen die Längen und Flächenmaße bestimmt (Fläche über Anzahl Pixel und Länge über Skelettierung) und mit klassenspezifischen Schwellwerten verglichen. Unterschreiten die Objekte die Schwelle, wird die Detektion im Logit-Raum des Detektors unterdrückt, um eventuell die zweitbeste Klassenhypothese aktivieren zu können (Abbildung 39 (d)).

3. Klassenfilterung

Nach Schritt zwei wird die Logit-Ausgabe des Netzwerks in eine Maximum Auswahl konvertiert, um pro Bildpunkt genau eine Klassenhypothese zu erhalten. Diese Ausgabe wird in Schritt 3 erneut mit den Schwellwerten aus Schritt 2 gefiltert (Abbildung 39 (e)).

4. Mehrheitsentscheid

In einem letzten Schritt werden zusammenhängende Detektionen von verwandten Klassen (z.B. Naht, EFLI, sanierter Riss) nach dem Mehrheitsentscheid zu einer Klassenentscheidung zusammengeführt.

Durch die Anwendung der Heuristik kann die Detektion des Detektors im Vergleich zur menschlichen Ground Truth noch einmal verbessert werden. Zwar hat die Heuristik kaum oder nur geringen Einfluss auf die mIoU, kann dafür jedoch den Occ-Score noch einmal deutlich steigern:

Tabelle 13: Verbesserung der Bewertungsmaße durch Einsatz der Nachverarbeitung der Detektionsergebnisse.

| | mIoU | Occ-Score |
|------------|------|-----------|
| Ohne Logik | 0.65 | 0.41 |
| Mit Logik | 0.65 | 0.57 |

7.3 Umwandlung von Flickstellen

Die Detektion von Flickstellen ist besonders bei sehr großen Flickstellen ein nicht triviales Problem. Ohne Kenntnis des Streckenkontexts lässt sich für ein Einzelbild in vielen Fällen nicht zweifelsfrei erkennen, welche Oberflächenart der "normalen" Oberfläche zuzuordnen ist und welche die eigentliche Flickstelle darstellt. Um diesen Umstand Rechnung zu tragen und um Probleme durch fehlerhaftes Labeling zu reduzieren, wird der Schadstellendetektor nur auf dem eigentlichen Übergangsbereich zwischen "normaler" Oberfläche und Flickstelle trainiert. Der so trainierte Detektor ordnet entsprechend bei der Detektion auch nur diesen Übergangsbereich den jeweiligen Flickstellenklassen zu (Abbildung 40).

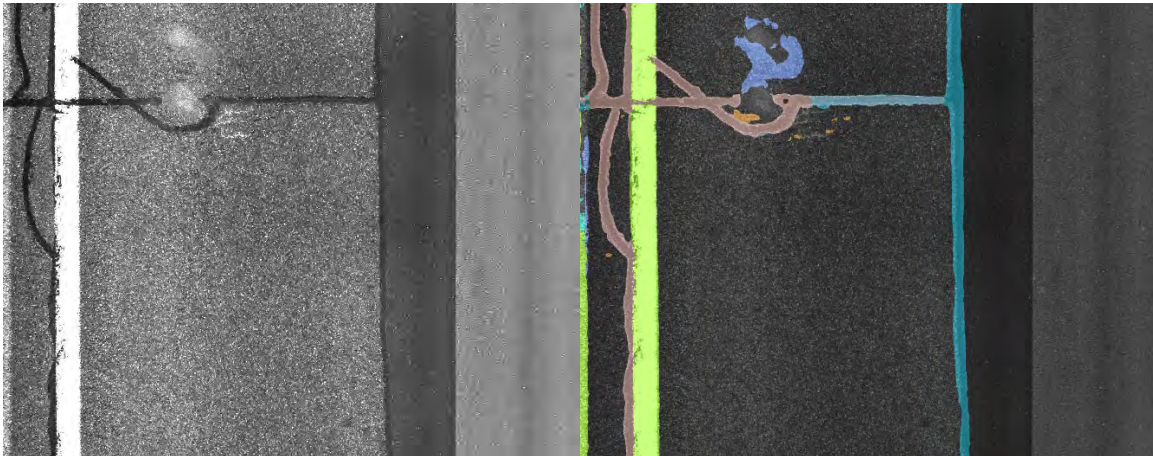


Abbildung 40: Detektion von Flickstellen über deren Übergangsbereich. links - Orthobild, rechts - Ausgabe

Um die korrekten Maße von Flickstellen bestimmen zu können, müssen diese Umrissdetektionen wieder zu Flächen konvertiert werden. Für diese Aufgabe werden zwei unterschiedliche Ansätze verfolgt. Zum einen die Umrechnung auf Grundlage des Umrisses und zum anderen die Umrechnung auf Grundlage der Oberflächenstruktur.

7.3.1 Bestimmung der Fläche über den Umriss

Wenn ein geschlossener Umriss einer Flickstellenklasse gefunden werden kann, lässt sich die Fläche der Flickstelle eindeutig bestimmen. Dieser Fall tritt häufiger bei aufgelegten Flickstellen auf, deren räumliche Ausdehnung in der Regel gering ausfällt. Auch bei kleineren Flickstellen kann es jedoch vorkommen, dass diese am Rand der Straße liegen oder von anderen Klassendetektionen "durchschnitten" werden. Aus diesem Grund wird die Detektion von Flickstellen vor der Auswertung durch morphologische Operationen (Schließen und Dilatation) geglättet und vergrößert. Außerdem werden Flickstellen durch Klassen, welche außerhalb der Straße auftreten (Bankett, Biomasse, Pflaster, Straßenablauf, Borde, Schacht), und von dem Bildrand zusätzlich begrenzt.

7.3.2 Bestimmung der Fläche über die Oberflächenstruktur

Bei Flickstellen, welche sich über eine größere Fläche erstrecken, ist eine Bestimmung der Fläche über den Umriss in vielen Fällen nicht möglich. Dies ist beispielsweise der Fall, wenn auf dem aktuell auszuwertenden Oberflächenbild nur ein sich über die ganze Bildausdehnung erstreckendes Fugenband zu erkennen ist (Abbildung 40). In diesem Fall kann der in Abschnitt 6.3 beschriebene Clusterer von Oberflächenstrukturen zum Einsatz kommen, welcher unterschiedlichen Strukturen jeweils eigene Merkmalsvektoren zuweist. Auf diese Weise kann das Oberflächenbild anhand der Merkmalsvektoren in Bereiche mit ähnlicher

Oberflächenstruktur geclustert werden. Durch Bestimmung der Merkmalsvektoren aus den vorherigen und folgenden Oberflächenbildern lässt sich der Bereich der Flickstelle eindeutig identifizieren und somit deren Fläche bestimmen.

7.3.3 Bestimmung von Flächenmaßen

Die von dem Schadens- und Objektdetektor produzierten Objektkarten ermöglichen eine nahezu pixelgenaue Zuordnung der Klassenlabel. Daher erfolgt die Bestimmung von Flächenmaßen durch simple Summierung der jeweiligen Klassenpixel.

Eine Ausnahme bildet der sogenannte Flächenausgleich bei der Umsetzung der Flickstellenbewertung des deutschen Standards. Auch hierfür ist die Umsetzung jedoch einfach möglich. Durch die Kenntnis der Fläche der Flickstellen kann diese ins Verhältnis zu der durch das Auswerteraster abgedeckten Fläche gesetzt werden. Entsprechend des ermittelten Quotienten wird die Anzahl der "aktivierten" Auswerteraster gleichverteilt reduziert.

7.3.4 Bestimmung von Längenmaßen

Längenmaße müssen von der Breite des detektierten Objekts unabhängig betrachtet werden. Außerdem können insbesondere Risse eine stark nicht linear ausgeprägte Form aufweisen, sodass eine Bestimmung der Länge eines Objekts über deren Außenkontur ebenfalls ausscheidet.

Für die Bestimmung von Längenmaßen wird daher auf einen Skelettierungs-/Thinningalgorithmus zurückgegriffen [Chen und Hsu, 1988], der Binärmasken auf 1-Pixel-breite Strukturen reduziert. Bei diesem Algorithmus wird die Breite iterativ reduziert, sodass schlussendlich nur noch die "Mittellinie" übrigbleibt (Abbildung 41). Wenn die Mittellinie extrahiert wurde, kann durch Aufsummieren der Pixel direkt die Länge des Objekts ermittelt werden.

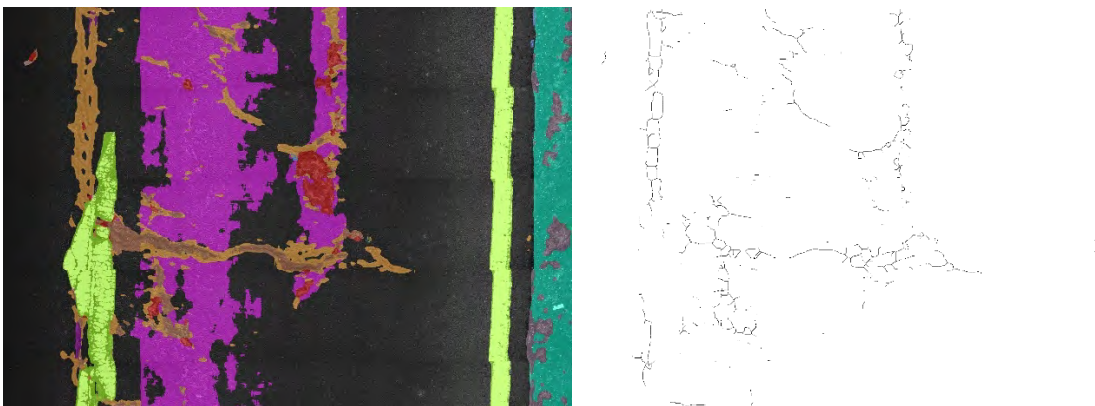


Abbildung 41: Diese Abbildung verdeutlicht die Bestimmung von Längenmaßen durch einen Skelettierungsalgorithmus. Auf der Rissklasse (Orange im linken Bild) wird eine Skelettierung durchgeführt, welche die Mittellinie des Rissverlaufs extrahiert, wodurch eine Ermittlung der Länge ermöglicht wird.

7.3.5 Detektion von Netzkissen

Die Detektion von Netzkissen erfolgt ebenfalls auf der Grundlage des Skelettierungs-/Thinningalgorithmus. In diesem Fall wird das resultierende Mittellinienbild jedoch auf Kreuzungspunkte untersucht. Wenn diese gefunden werden, teilt sich ein Riss in mehrere Teilrisse auf. Ab einem bestimmten Länge-zu-Kreuzungspunkt-Verhältnisses, welches frei konfiguriert werden kann⁶, wird ein Riss als Netzriss klassifiziert und fließt entsprechend in die Bewertung ein.

7.3.6 Lokalisation von Schäden auf Betonplatten

Um das Training für die Betondetektion zu verbessern, wurden unter 3.3.1 verschiedene Betonklassen, die eine ähnliche optische Ausprägung besitzen, aber in den Schadenskatalogen auf unterschiedliche Schadenklassen abgebildet werden, zu gemeinsamen Labelklassen vereint. Um nach der Detektion dieser Labelklassen dann wieder auf die entsprechende Schadenklasse schließen zu können, müssen die Schäden in Bezug auf ihre Position innerhalb der Betonplatte analysiert werden.

Im ersten Schritt müssen dazu die Betonplatten selbst gefunden werden. Dazu wird auf der Detektionsklasse Plattengrenze eine Hough-Linientransformation für vertikale und für horizontale Linien durchgeführt (Abbildung 42). Zusammenhängende Bereiche auf der inversen Linienkarte bilden anschließend die eigentlichen, zusammenhängenden Betonplattenbereiche.

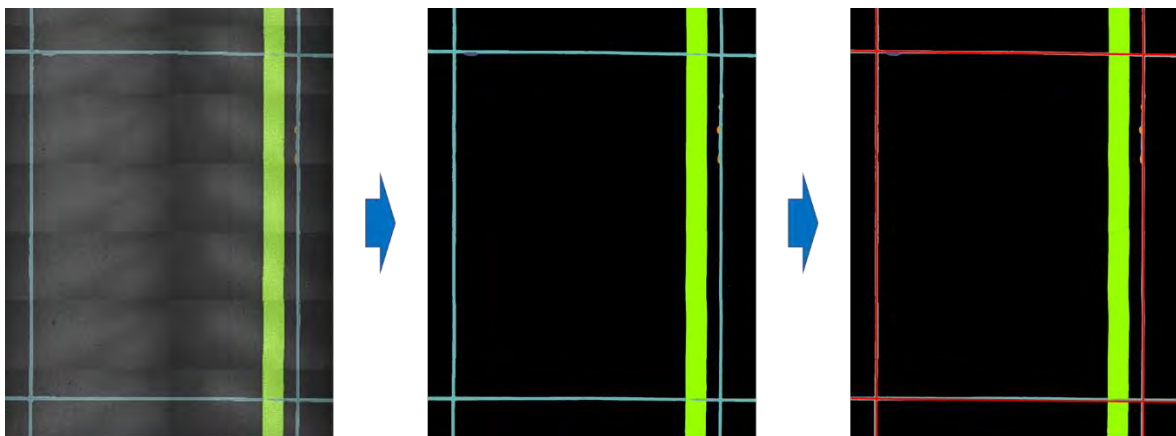


Abbildung 42: Finden von Betonplatten. Um Betonplatten finden zu können, wird auf der Plattengrenzendetektion (Mitte - Türkis) eine Hough-Linientransformation durchgeführt, um Horizontale und Vertikale Plattengrenzen bestimmen zu können (rechts).

⁶ Auf einer manuell mit Netzkissen gelabelten Teststrecke, ergab sich für ein Kreuzungspunkt-Verhältnis von 3 die größte Übereinstimmung zwischen manueller und automatischer Detektion.

Die Entscheidung, ob ein Schaden einer Rand- oder Eckkategorie zugerechnet wird, erfolgt anschließend durch Auswertung der Schnittbereiche zwischen den Plattengrenzen-Linien und den Schäden. Berührt ein Schaden eine horizontale und eine vertikale Begrenzungslinie einer Platte (und erfüllt die länderspezifischen Maßanforderungen), wird er einer Eckkategorie zugeordnet. Berührt er hingegen nur eine Kante (und erfüllt die länderspezifischen Maßanforderungen), wird er einem Kantenschaden zugeordnet.

8 Demonstrator

Für die Integration der im Projekt entwickelten Teilfunktionalitäten war die Entwicklung einer graphischen Benutzeroberfläche (GUI) vorgesehen. Mit dieser GUI sollte zum einen die Funktionsfähigkeit der Komponenten demonstriert und zum anderen Wege für die Integration der Komponenten in die klassische Zustandsbewertung aufgezeigt werden.

Um den Funktionsumfang der GUI festzulegen und die Erwartungen und Vorstellungen der Auftragnehmer und Auftraggeber in Einklang zu bringen, wurde im Rahmen des Projekts ein User-Interface Workshop durchgeführt. In diesem wurden grundlegende Anforderungen, User-Stories und auch erste Mock-ups für die Benutzeroberfläche erstellt (Abbildung 43). Darauf folgend wurde der Funktionsumfang über Telefonkonferenzen weiter verfeinert und zwischen allen Projektbeteiligten abgestimmt. Die vollständige Spezifikation würde den Rahmen dieses Berichts sprengen und ist daher nicht Teil des Berichts.

Dabei stellte sich heraus, dass die notwendigen Anforderungen an die GUI erheblich höher liegen, als dies im Projektantrag vorgesehen und personell eingeplant war. Entsprechend mussten Anpassungen vorgenommen werden, um der Benutzeroberfläche im Projekt genügend Raum zu verschaffen.

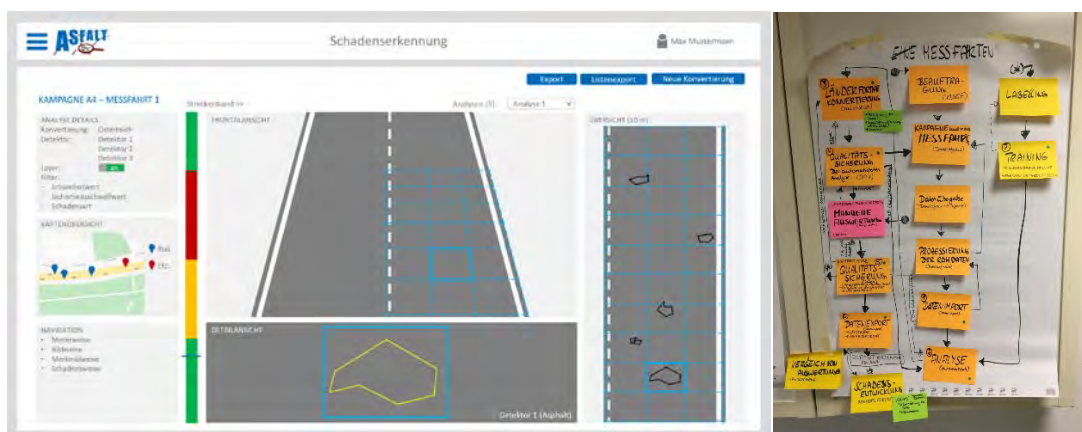


Abbildung 43: Beispiele für die während des GUI-Workshops erarbeiteten Mock-Ups und Prozessabläufe.

Die während des Workshops erarbeiteten Mock-ups und User-Stories haben für eine sehr zielgerichtete Entwicklung gesorgt, so dass viele Anforderungen bereits im Vorfeld geklärt

werden konnten. Bei den anschließenden Benutzertest wurden daher nur wenige Änderungen gewünscht, die einen Eingriff in die grundlegenden Abläufe notwendig gemacht haben.

8.1 Rahmenbedingungen und technische Anforderungen

Eine wichtige "nicht funktionale" Anforderung der Auftraggeber ist die Ausführbarkeit der ASFaLT-GUI unter Windows. Um auch der Softwareinfrastruktur des maßgeblich an der Entwicklung der GUI beteiligten TU Ilmenau Rechnung zu tragen, welche fast ausschließlich auf Linux als Betriebssystem aufsetzt, sollten von der GUI beide Betriebssysteme gleichermaßen unterstützt werden. Die Umsetzung erfolgte daher auf Grundlage der Programmiersprache Python und setzt für die GUI-Elemente im Wesentlichen auf der PySide Bibliothek auf.

Neben der eigentlichen GUI, welche für die Interaktion mit dem Benutzer erforderlich ist, besteht die Softwareinfrastruktur des ASFaLT-Projekts aus zwei weiteren Komponenten, welche für die Bereitstellung der Funktionalität erforderlich sind. Dies ist zum einen eine MySQL-Datenbank mit einer dazugehörigen Serverapplikation. In der Datenbank werden Befahrungen, Klassifikationsergebnisse und zusätzliche Metadaten verknüpft, um sie in der GUI visualisieren zu können. Zum anderen ist dies eine als ASFaLT-Server bezeichnete Software-Komponente, welche für die Ausführung der eigentlichen Segmentierung und Bewertung der Straßendaten erforderlich ist. Auch diese Komponente basiert auf Python und setzt mit der Python-Bibliothek Tensorflow auf eine Bibliothek, welche ebenfalls einen Einsatz unter den Betriebssystemen Windows und Linux ermöglicht.

8.2 Benutzeroberfläche

Um einen Eindruck von dem im Projekt umgesetzten Demonstrator zu vermitteln, sollen im Folgenden die wesentlichen Komponenten der Benutzeroberfläche vorgestellt werden.

8.2.1 Messfahrtenübersicht

Das zentrale Element der Benutzeroberfläche, von welcher aus alle Teilfunktionalitäten gestartet werden können, bildet die Messfahrtenübersicht. In dieser Ansicht werden alle importierten und gegebenenfalls bereits analysierten Messfahrten mit den wesentlichen Informationen dargestellt (Abbildung 44). Aus dieser Ansicht ist es weiterhin möglich, neue Messfahrten zu importieren, Detektionsergebnisse von einzelnen Messfahrten aufzurufen oder den Vergleich zwischen mehreren Befahrungen / Auswertungen zu starten.

| Name | Beschreibung | Jobname | Standort | Straße | von | bis | Fortschritt | Status |
|--|--------------|------------------|-----------|----------------|-----|-----|-------------|--------|
| A02-69R1-HAST Edlitz_KM0,000_KM0,000_210421_0939 | | | | A02-69 | 0 | 0 | 100% | |
| Folder | | | Unbekannt | | | | 100% | |
| G05_21022415_T3_M_conc | Beton | Initialer Import | 050000 | Bundesautobahn | | | 100% | |
| G05_21022415_T3_M_mod | | Initialer Import | 050000 | Bundesautobahn | | | 100% | |
| G05_X2141500_TP3_mod | | Initialer Import | 050000 | Bundesautobahn | | | 100% | |

Abbildung 44: In der Messfahrtenübersicht werden die importierten Messfahrten mit einigen zusätzlichen Informationen angezeigt. Über die Statusampel ist ersichtlich, ob eine Messfahrt bereits erfolgreich analysiert wurde. Über einen Doppelklick auf eine Befahrung wird die Messfahrtenanalyse gestartet. Wenn über die Checkboxes zwei Messfahrten selektiert wurden, wird der „Messfahrten vergleichen“-Button aktiviert, über den die Vergleichsansicht gestartet werden kann.

8.2.2 Datenimport

Der Import von neuen Daten wird über die Schaltfläche *Neue Messfahrt* gestartet. Darauf startet der Dialog für den Import von Daten, welcher in Abbildung 45 dargestellt ist. In dieser Ansicht können, die in den Ländern Österreich und Deutschland üblichen, Datenformate importiert werden. Zusätzlich ist auch der Import von reinen Bilddaten möglich, um ein möglichst breites Spektrum an Anwendungsfällen abzudecken. Außerdem kann die Messfahrt mit einem Namen und einer Beschreibung versehen und die initiale Einstellung für die Detektoren festgelegt werden.

Abbildung 45: Import Dialog - Über den Button „Öffnen“ wird das Verzeichnis mit den Front- und Orthobildern der Befahrung ausgewählt. Die Textfelder werden durch die aus dem Verzeichnis extrahierten Informationen vorausgefüllt.

Nachdem die Messfahrt über einen Klick auf *Analyse starten* in die Datenbank integriert wurde, erkennt die Server-Komponente die neue Messfahrt und beginnt automatisch mit der Analyse. Der Fortschritt ist in der Projektübersicht über die dargestellte Ampel möglich.

8.2.3 Messfahrtenanalyse

Wenn eine Messfahrt vollständig analysiert wurde, kann die Ansicht einer einzelnen Messfahrt durch einen Doppelklick auf die gewünschte Messfahrt initiiert werden. In der sich öffnenden Ansicht (Abbildung 46) kann die Messfahrt in verschiedenen Detailstufen analysiert werden. Neben allgemeinen Informationen der Messfahrt können, für alle in der Messfahrt gefundenen Schadenstypen, Streckenbänder aktiviert werden, welche einen Überblick über die Häufigkeit und die Verortung entlang der Messfahrt geben. Über die an den Streckenbändern angezeigten Pfeilsymbole kann jeweils zwischen der vorherigen und der nächsten Instanz eines Schadens entlang der Messfahrt navigiert werden.

Für den aktuell gewählten Streckenabschnitt wird jeweils das dazugehörige Front- und Orthobild angezeigt. Zusätzlich wird in einer Detailansicht ein kleinerer Bereich der Abschnittsübersicht vergrößert dargestellt. Der in der Detailansicht vergrößerte Bereich kann durch Klicken mit der Maus in der Abschnittsübersicht frei gewählt werden. Sowohl in der Abschnittsübersicht als auch in der Detailansicht wird das jeweiligen Orthobild des aktuellen Streckenabschnitts dargestellt. Zusätzlich zu dem Bewertungsraster werden die detektierten Schadens- und Objektinstanzen in unterschiedlichen Farben als Überlagerung dargestellt. Dabei kann die Auswahl der angezeigten Klassen eingegrenzt werden und auch der Grad der Überlagerung über Schieberegler gewählt werden, um die Detektion und das zugrundeliegende Straßenbild besser beurteilen zu können.

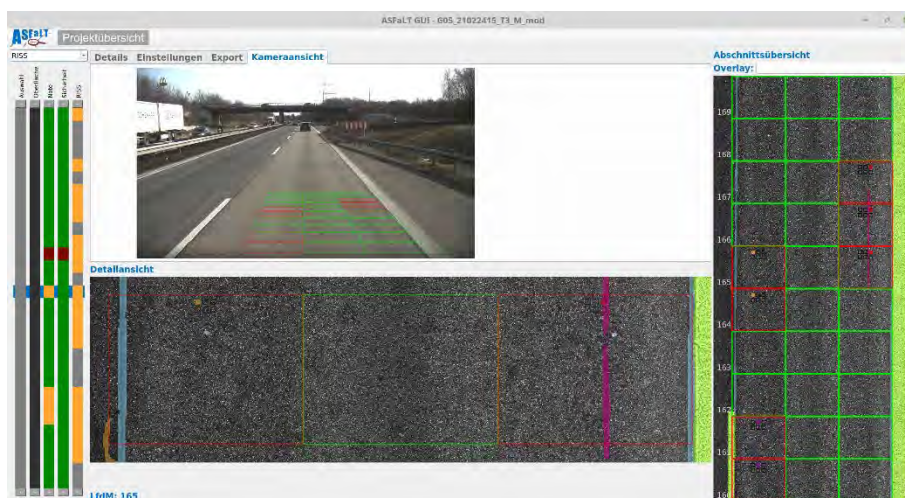




Abbildung 46: Darstellung der Visualisierungen von Befahrungen, welche nach deutschem Standard (oben) oder nach österreichischem Standard (unten) ausgewertet wurden. Die Streckenbänder auf der jeweils linken Seite geben einen ersten Eindruck über die Verteilung verschiedener Schadens- und Objektklassen innerhalb der Messfahrt. Durch Auswahl eines Streckenabschnitts werden die zugehörige Front- und Orthobilder geladen. Für eine bessere Beurteilung steht zusätzlich eine Detailansicht zur Verfügung, welche Bereiche der Abschnittsübersicht vergrößert darstellt. Die Einstellungen für die Detektion und Länderformatkonvertierung kann für einzelne Bilder oder ganze Abschnitte vorgenommen werden.

8.2.4 Auswertungsvergleich

Messfahrten, die mit unterschiedlichen Einstellungen oder Detektoren bewertet wurden, können mithilfe des Auswertungsvergleichs gegenübergestellt werden. Dazu werden in der Messfahrtenübersicht zwei zu vergleichende Messfahrten selektiert und anschließend der Button *Messfahrten vergleichen* ausgewählt.

Die sich öffnende Ansicht (Abbildung 47) besitzt die gleichen Elemente wie die Messfahrtenanalyse. Allerdings werden hier zwei Befahrungen nebeneinander synchronisiert dargestellt. Das standardmäßig angezeigte Streckenband zeigt, an welchen Stellen es in den Bewertungen Abweichungen gibt. Zusätzlich können auch in dieser Ansicht Streckenbänder für alle detektierten Klassen angezeigt werden. Diese werden dann jeweils getrennt für beide Auswertungen angezeigt. Die Elemente Kameraansicht, Abschnittsübersicht und Detailansicht verfügen über die gleichen Optionen wie in der Messfahrtenanalyse und können für die bessere Beurteilung angepasst werden.

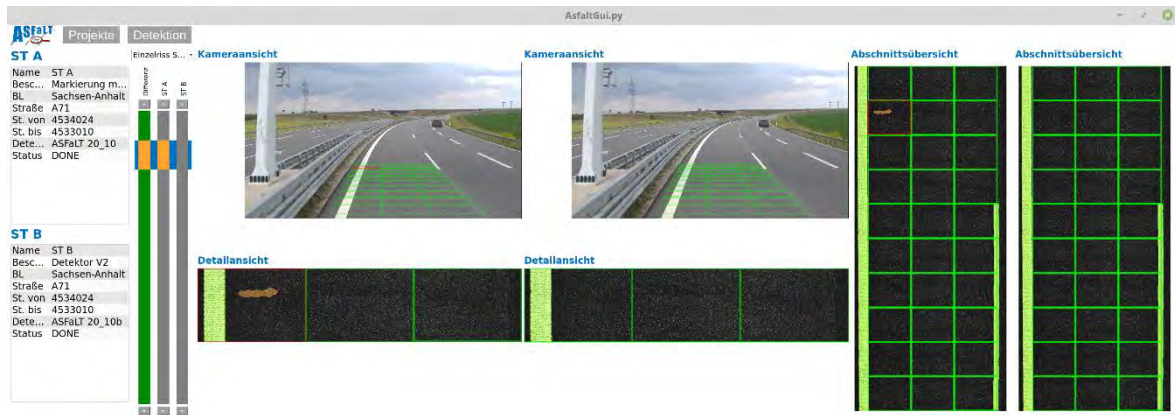


Abbildung 47: Messfahrtenvergleich - Zwei Auswertungen von Messfahrten können in dieser Ansicht vergleichend dargestellt werden. Dazu sind die Bedienelemente synchronisiert und in den Streckenbändern wird jeweils der Unterschied zwischen den Auswertungen hervorgehoben.

8.3 Umsetzung der User-Stories

Für eine genauere Beurteilung des Standes der Umsetzung wird in den nachfolgenden Abschnitten eine Gegenüberstellung aller User-Stories mit der Umsetzung durchgeführt.

Die User-Stories werden dabei getrennt nach den Kategorien *Basic* und *Nice-to-have* betrachtet, welche den User-Stories in Abstimmung mit allen Projektbeteiligten zugewiesen wurden.

8.3.1 Basic User-Stories

Wie aus Tabelle 14 ersichtlich ist, sind alle User-Stories fast vollständig umgesetzt. Damit ist der Import, die Verarbeitung und der Export der Daten über die GUI sichergestellt. Außerdem kann die automatische Detektion über die GUI beurteilt werden. Nicht mehr im Projektzeitraum umgesetzt werden konnte die Anzeige von Umentscheidungen, von erweiterten Markern und der Export nach Excel.

Tabelle 14: Überblick über die Umsetzung der Basic User-Stories.

| US | Beschreibung | Umsetzung |
|----|--|------------|
| 3 | Ich, Datenimporteur, möchte Front- und Orthobilder inklusive der Metadaten importieren, um Dateneinheit im Analysetool herzustellen. Beim Import wird der übergeordnete Ordner ausgewählt, die Daten werden bezüglich einer Messfahrt hochgeladen. | Umgesetzt. |
| 4 | Ich, Datenimporteur, möchte eine leere XML-Datei importieren, um sie automatisch auswerten zu lassen. | Umgesetzt. |
| 5 | Ich, Datenimporteur, möchte meine Standard- und Massendaten konfliktfrei importieren, um Informationsverlust zu vermeiden. | Umgesetzt. |
| 6 | Ich, Datenimporteur, kann mein länderspezifisches Format hochladen, um die Daten weiterzubearbeiten. Dafür benutze ich eine | Umgesetzt. |

| | | |
|------------------|--|---|
| | konfigurierbare Schnittstelle (z.B. Pfad). Um eine Analyse hinzuzufügen, kann die jeweilige Messfahrt angelegt werden oder zu einer Messfahrt eine Analyse über die jeweiligen (+)-Zeichen angelegt werden. | |
| 8/ 10 | Ich, Datenimporteur, möchte die importierten Daten analysieren lassen (automatisch), um die Strecke auswerten zu können. Ich, Datenimporteur, starte Analyse mit Detektor, um automatisch eine Schadensstellendetektion zu erhalten. | Umgesetzt. |
| 14 | Ich, Auswerter, möchte regelmäßig / schnellstmöglich von den neuen Trainingsdaten „der Anderen“ profitieren, um Auswertung und Aufwand zu verbessern. | Umgesetzt. Es kann immer auf eine aktuelle Liste von Klassifikatoren zugegriffen werden. |
| 15 | Ich, Konvertierer, möchte sicherstellen, dass die Ableitung des Länderstandards korrekt funktioniert, um regelkonforme Ergebnisse zu erzeugen. | Umgesetzt. Die Detektionen werden auch im länderspezifischen Raster angezeigt. |
| 16/ 21 | Ich, Benutzer, kann Konfigurationsdaten für die Konvertierung vorgeben bzw. einstellen (z.B. Schwellwerte). Ich, Qualitätssicherer, lege den Unterschied zwischen Netz- und Einzelrissen fest, um einen guten Schwellwert für alle Unterscheidungen zu finden. | Die Einstellung von Parameter ist über die GUI und über ein Skript möglich. |
| 17/ 31/ 33 | Ich, Qualitätssicherer, sehe sofort, wie sicher die Länderkonvertierung die Ergebnisdaten erkannt hat. Es können sichere Ergebnisse von unsicheren unterschieden werden. Ich, Benutzer, sehe die Konfidenzintervalle pro Bild und Schaden, um die Sicherheit der Erkennung bewerten zu können. Ich, Qualitätssicherer, möchte die Sicherheit der automatischen Detektion interpretieren können, um den Anteil der erforderlichen manuellen Nacharbeit korrekt einzuschätzen. | Umgesetzt. Sicherheiten werden als Streckenband angezeigt. |
| 18/ 22 | Ich, Qualitätssicherer, bekomme die Schäden visualisiert, um die Ergebnisse zu validieren. Jeder Schaden wird bzgl. des Schadenkatalogs identifiziert und klassifiziert angezeigt. Zudem wird das Länderraster angezeigt. Ich, Qualitätssicherer, überprüfe die Qualität der automatischen Auswertung, um Fehler beim Import / beim Labeling festzustellen. | Umgesetzt. Schäden werden örtlich aufgelöst dargestellt und in länderspezifisches Raster überführt. |
| 19/ 23 | Ich, Qualitätssicherer, bekomme einen schnellen Überblick über die Unsicherheiten der Messfahrt, um die Qualität der Analyse bewerten zu können und Sequenzen für ein manuelles Labeling festzulegen. Ich, Qualitätssicherer, schaue mir die automatische Auswertung an, um zu entscheiden, was manuell nachausgewertet werden muss. | Unsicherheiten werden angezeigt und Bereiche können für manuelles Labeling ausgewählt werden. |

| | | |
|-----------|---|---|
| 20/ 35 | Ich, Qualitätssicherer, verschaffe mir eine schnelle Übersicht über die Marker, um die Plausibilität der Belagsarten und Schäden zu prüfen. Ich, Qualitätssicherer, bekomme einen schnellen Überblick über die erkannten Belagsarten, um schnell zu erkennen, ob der richtige Detektor verwendet wurde. | Belagarten werden als Streckenband angezeigt. Sonstige Markerinformationen werden nicht visualisiert. |
| 24 | Ich, Qualitätssicherer, möchte mir die Qualitätsmerkmale und Schadensstatistiken in unterschiedlichen Detaillierungsgraden anzeigen lassen. Dazu kann ich mir unterschiedlich dargestellte Streckenbänder anzeigen lassen, mit verschiedenen Informationen. | Umgesetzt. Streckenbänder können für allgemeinen Zustand und für einzelne Schadensklassen angezeigt werden. |
| 25 | Ich, Qualitätssicherer, kann verschiedene Analysen zu einer Messfahrt auswählen, um mir nacheinander Analysen anschauen zu können. | Umgesetzt. |
| 26 | Ich, Qualitätssicherer, exportiere die Strecke / Daten / Streckenabschnitt, damit sie vom Auswerter nachausgewertet werden kann. | Daten können in länderspezifischen Formaten exportiert werden. |
| 27 | Ich, Benutzer, kann eine Liste ausdrucken (z.B. Excel), um die nachzuprüfenden Schäden der manuellen Auswertung zur Verfügung zu stellen. | Nicht umgesetzt. |
| 28 | Ich, Qualitätssicherer, ändere die Konfiguration der Länderformatkonvertierung, um diese neu auszuführen (z.B. Schwellwert). Dazu verwende ich die Schaltfläche „Konvertierung“. Für die Konvertierung kann ich das gewünschte Länderformat auswählen. | Die Konfigurationen für die Länderformatkonvertierung können angepasst werden. |
| 29 | Ich, Qualitätssicherer, kann den Detektor und den Bereich bzw. Bilder auswählen, die in der Analyse verwendet werden, um eine neue Analyse anzustoßen. | Detektoren und Einstellungen können für Bereiche festgelegt werden. |
| 32 | Ich, Qualitätssicherer, kann die Fahrbahnbegrenzungen erkennen und korrigieren, um die Qualität zu verbessern (Raster). | Fahrbahnbegrenzung wird automatisch gesetzt kann allerdings nicht angepasst werden. |
| 36/ 41 | Ich, Qualitätssicherer II, vergleiche die importierten Daten mit der ursprünglich automatischen Auswertung, um Schwächen bei automatischen Auswertungen, aber auch bei manuellen Auswertungen, zu finden. Ich, Qualitätssicherer II, möchte zwei Auswertungen vergleichen können, um... | Umgesetzt. Vergleich von zwei Befahrungen ist möglich. Sowohl länderspezifische Raster als auch Details werden angezeigt. |
| 37/ 38 | Ich, Qualitätssicherer II, suche nach Schwächen in der automatischen Detektion, um zu entscheiden, dass neu gelabelt werden muss. Ich, Qualitätssicherer II, suche nach Schwächen in der manuellen Auswertung, um zu entscheiden, dass Auswerter nachgeschult werden muss. | Über Vergleich von Befahrungen möglich. Streckenband zeigt Regionen mit Unterschieden. |
| 40 | Ich, Qualitätssicherer II, sehe die Umentscheidungen der Auswerter, die Häufungen der Schäden, weiß die neuen Strecken | Detektoren und Schadenshäufigen können angezeigt werden. |

| | | |
|-------|--|--|
| | und die Ausbesserungen, kenne die Detektoren, um bewerten zu können. | Ummentscheidungen werden nicht visualisiert. |
| 42 | Ich, Qualitätssicherer II, kann einen Differenzreport erstellen, in dem die Unterschiede zwischen den zu vergleichenden Analysen tabellarisch dargestellt werden. | Ein tabellarischer Export ist nicht möglich. |
| 44 | Ich, Auswerter, möchte ohne viel IT-Wissen selbständig trainieren können, um die automatische Auswertung zu verbessern. | Das Training kann über einen Trainingsscript weitestgehend automatisch erfolgen. |
| 45 | Ich, Trainer, möchte angezeigt bekommen, ob die Qualität besser geworden ist, um eine Entscheidungsgrundlage zu haben (z.B. durch Scores) ? universell, nicht länderspezifisch. | Bewertung ist über Occurrence-Score umgesetzt. |
| 46/47 | Ich, Qualitätssicherer II, exportiere die Daten, damit diese von anderen Systemen weiterverarbeitet werden können. Ich, Datenexporteur, möchte regelkonforme Ausgabedaten und originale Detektionsergebnisse exportieren, um möglichst keine Informationen zu verlieren. | Die verarbeiteten Daten können über die MySQL-Datenbank und Detektionskarten weiterverarbeitet werden. |

8.3.2 Nice-to-have User-Stories

Die Nice-to-have User-Stories standen bei der Umsetzung der GUI nicht im Fokus. Dennoch konnten bereits einige User-Stories erfolgreich umgesetzt werden, da sie sich sinnvoll in die Umsetzung der Grundfunktionalitäten integriert haben.

Ein vollständiger Überblick über den Stand der Umsetzung der Nice-to-have User-Stories ist der Tabelle 15 zu entnehmen.

Tabelle 15: Überblick über die Umsetzung der Nice-to-have User-Stories.

| US | Beschreibung | Umsetzung |
|----|---|--|
| 1 | Ich, Datenimporteuer, lege eine neue Kampagne an, indem ich auf die Schaltfläche „Neue Kampagne“ klicke (vgl. Abbildung 2). Dies trifft z.B. für die Analysen zu, wenn noch keine Analyse für den Streckenabschnitt vorliegt. Anschließend können die jeweiligen Informationen zu einer Kampagne eingegeben werden. | Keine Unterstützung von Kampagnen umgesetzt. |
| 2 | Ich, Datenimporteuer, lege eine neue Messfahrt zu einer bestehenden Kampagne an und nutze dafür das Aktionssymbol (+) bei der jeweiligen Kampagne. Beim Anlegen der Messfahrt, findet auch der eigentliche Datenimport statt. Auftretende Fehler beim Import werden beschrieben. | Keine Unterstützung von Kampagnen. Datenimport und Fehlerbehandlung umgesetzt. |
| 7 | Ich, Importeur, kann den Detektor auswählen, um die Analyse zu konfigurieren (und Bereich / Bilder). | Auswahl von Bereichen und Bildern möglich. |
| 9 | Ich, Auswerter, möchte sicherstellen, dass ich den richtigen Detektor mit den richtigen Parametern aktuell nutze, um eine bestmögliche Klassifikation zu generieren. | Umgesetzt |

| | | |
|----|---|---|
| 11 | Ich, Datenimporteuer, kann die Analyse terminlich definieren und bekomme eine zeitliche Abschätzung der Dauer, um Analysen auch nachts laufen zu lassen. Dazu wird eine Übersicht über geplante Analysen unterstützt. | Fortschritt wird prozentual angegeben. Noch Abzuarbeitende Jobs werden in der Jobübersicht angezeigt. |
| 13 | Ich, Admin, kann die Anwendung beliebig skalieren, um die Performanz zu optimieren. Ich weiß auch, dass die Performanz durch die Aufteilung der Daten optimiert wird (Anzahl der Grafikkarten). | Einzelne Anwendung unterstützt nur eine Grafikkarte. |
| 30 | Ich, Benutzer, sehe die importieren Dateien, um eine Übersicht zu haben und kann einen Einstieg über Dateien und / oder über unsichere Riss- / Schadenserkenung nehmen (Zuordnung zu Dateien und Zuordnung der Bereiche zu Dateien ist erforderlich). | Über Streckenbänder umgesetzt. |
| 39 | Ich, Qualitätssicherer II, möchte Zugriff auf alle prozessrelevanten Informationen (z.B. Chronologie) mit dem Ziel, eine umfassende Perspektive zu haben. | Chronologie wird nicht unterstützt. |
| 43 | Ich, Qualitätssicherer II, vergleiche 2 länderspezifische Auswertungen, um deren Qualität zu bewerten. | Vergleiche von zwei Auswertungen mit jeweils dem gleichen länderspezifischen Standard ist umgesetzt. |
| 48 | Ich, Datenexporteur, führe den Datenexport in länderspezifische und Einheitsformate durch, um Daten an Kunden zu liefern (für Reports). | An Kundenwünsche anpassbare Reports sind nicht im Projekt vorgesehen. |

8.4 Benutzertests

Um die Nutzbarkeit der grafischen Oberfläche zu testen und verbessern zu können, waren im Projekt Benutzertest vorgesehen, bei welchen unterschiedliche Operateure den Demonstrator mit realen Testdaten testen sollten.

Die Benutzertests wurden jeweils von allen Auftragnehmern durchgeführt. Zusätzlich erfolgten die Benutzertests durch Vertreter der BASt, so dass auch die Sicht der Auftraggeber bei den Benutzertest abgebildet werden konnte. Die Tests wurden im Projekt in zwei Zeiträumen durchgeführt, um zwischen dem ersten und dem zweiten Testzeitraum Anpassungen an dem Demonstrator vornehmen zu können. Nach dem ersten Testzeitraum (31. Mai bis 04. Juni 2021), wurde von allen Testern eine Liste von potenziellen Punkten zur Optimierung angefertigt.

Auf dieser Liste fanden sich vor allem Punkte für die Optimierung des Workflows, aber auch mit Bezug zur Performanz und Punkte mit Fehlern bei der Dateneingabe und dem Export in die Länderformate. Bis auf einige optische Anpassungen konnten in der darauffolgenden

Implementationsphase alle wesentlichen von den Testern angesprochenen Punkte umgesetzt werden.

Im zweiten Testzeitraum (25. August bis 01. September 2021) wies die Software daher deutlich mehr Einstellungsmöglichkeiten und Features auf, so dass die Software eine gute Analyse der automatisierten Analyse zulässt und den Export in die länderspezifischen Datenformate sicherstellt.

Es ist zu erwähnen, dass der Vergleich von Befahrungen nicht mehr rechtzeitig für die Benutzertests umgesetzt werden konnte. Diese Komponente ist daher zwar Teil der finalen Software, wurde jedoch im Rahmen der Benutzertest nicht hinsichtlich ihrer Nutzbarkeit unter realen Bedingungen überprüft.

9 Vergleich mit traditioneller standardisierter Bewertung

In diesem Teil des Berichts wird der Vergleich der Leistung des entwickelten Systems mit der klassischen, manuell ausgewerteten Befahrungen verglichen.

Hierfür mussten Testdaten, welche sich nicht mit der Menge der Trainingsdaten überschneiden, und die dazugehörigen Auswertungen akquiriert und anschließend von dem System ausgewertet werden. Die klassischen Auswertungen und die vom System automatisch erstellten Auswertungen wurden abschließend verglichen und bewertet.

9.1 Datengrundlage

Die für die Auswertung herangezogenen Daten sollten alle umgesetzten Länderstandards und auch alle im Projekt betrachteten Oberflächenarten abbilden. Als Testdaten wurden daher sowohl Befahrungen aus Deutschland als auch aus Österreich, jeweils mit den Oberflächenarten Asphalt und Beton, herangezogen. Um alle Testdaten aus den Ländern gleichermaßen mit jeweils einem Detektor für jede Oberflächenart verarbeiten zu können, mussten alle Bilder auf eine einheitliche metrische Größe skaliert werden. In diesem Fall stellten einige Oberflächenbilder von einem Kamerasystem, welches für die Befahrungen in Deutschland eingesetzt wurde, die Anforderungen an die obere Grenze der Auflösung. Da die Bilder nur mit einer Auflösung von 436 Pixel pro Quadratmeter aufgenommen wurden, wurde auch das KI-System nur auf eine Auflösung von 450 Pixel trainiert und angewendet, um übermäßiges Hochskalieren von Bilddaten zu vermeiden. Dies muss allerdings auch für die Auswertung berücksichtigt werden, da der KI damit teilweise fast nur 1/3 der Auflösung zur Verfügung steht, die bei der manuellen Auswertung genutzt werden konnte (Die Bilder mit der höchsten Auflösung erreichen 1200 Pixel pro Meter).

9.1.1 Testdaten aus Deutschland

Die Auswahl von Testdaten erfolgte für Deutschland über den Zugriff auf den IT-ZEB Server der BAST. Hierfür wurde von der BAST der Zugriff auf die Zustandsbefahrungen aus den Jahren 2015/16 und 2017/18 freigeschaltet, so dass sowohl auf Befahrungen von Bundesstraßen als auch auf Befahrungen von Autobahnen zugegriffen werden konnte.

Vorbemerkung

Wie sich bei der Bearbeitung dieses Teilarbeitspakets herausstellte, war ein massenhafter Download von hunderten von Testsequenzen vom IT-ZEB-Server nicht trivial und effizient möglich. Grund hierfür war, dass der IT-ZEB-Server keinen API für einen automatisierten Download von einer großen Menge an Daten anbietet. Daher mussten die Daten entweder manuell oder über automatisierte webbasierte Skripte heruntergeladen werden. Diese Art der Datenbeschaffung erwies sich als sehr zeitaufwändig und fehleranfällig. Aus diesem Grund besitzt die im folgenden Abschnitt vorgestellte Auswahl der Menge der Testdaten nicht in allen Aspekten die angestrebte Verteilung der Daten und spiegelt nicht in allen Aspekten die verhältnismäßige Verteilung der Daten (z.B. in Bezug auf die Gesamtmenge der Straßen in den einzelnen Bundesländern) wider.

Der Grund für die abweichenden Größenverhältnisse liegt in fehlerhaften Downloads, welche bei der Teilautomatisierung der Downloads entstanden sind⁷.

Auswahl der Testdaten

Da es eine sehr große Anzahl von möglichen Testbildern aus den Jahren 2015-2018 auf dem IT-ZEB-Server gibt, wurde für die Auswertung eine Vorauswahl getroffen, welche Art von Oberflächenbildern bewertet werden soll. Generell wurde entschieden, drei Kategorien von Oberflächenbildern zu beschaffen:

- Fehlerfreie Oberflächenaufnahmen, wie bereits im Ergebnisbericht 3.2 beschrieben
- Aufnahmen mit mittelmäßig vielen Schäden (Riss-Note zwischen 3.0 und 4.0)
- Aufnahmen mit sehr vielen Schäden (Riss-Note zwischen 4.5 und 5.0)

⁷ Die Teilautomatisierung simuliert über Skripte die auch durch einen realen Benutzer zu bewerkstellenden Auswahlaktionen auf der IT-ZEB-Webseite. Durch schwankende Internet-Konnektivität kann aber z.B. die Aktualisierung von Auswahlboxen für die Strecken variieren, so dass ggf. während der Automatisierung falsche Daten ausgewählt wurden. Diese fehlerhaften Downloads wurden im Nachgang gefiltert. Ein erneuter Download der fehlerhaften Daten konnte aus Zeitgründen nicht erfolgen.

Für diese drei Kategorien von Oberflächenbildern wurden jeweils 100-Meter-Abschnitte (im Folgenden als Sequenzen bezeichnet) aus Asphalt- und Betonaufnahmen der Bundesstraßen und Autobahnen selektiert, um eine möglichst hohe Variabilität der Daten zu gewährleisten. In einem internen Arbeitstreffen der Projektpartner (25.02.2021) wurde zusätzlich festgehalten, dass nur der rechte Fahrstreifen ausgewählt werden soll, da dieser typischerweise einer größeren Beanspruchung ausgesetzt ist.

Da diese Vorauswahl jedoch weiterhin sehr viel Daten beinhaltet, wurde darüber hinaus nur ein kleiner Prozentanteil aus dieser Vorauswahl geladen. Insgesamt wurde angestrebt pro Bundesland jeweils etwa 2% aller Aufnahmen aus diesen Kategorien zu laden. Hierbei wurde darauf geachtet, dass die 2% zufällig gezogen werden und sich aus der Gesamtstreckenlänge pro Kategorie pro Bundesland ergeben. Da die einzelnen Streckenabschnitte auf dem IT-ZEB-Server oft von sehr variabler Länge waren, wurde sich für die zufällige prozentuale Auswahl von Testdaten an der Gesamtlänge aller Strecken pro Bundesland und nicht an der Anzahl der Streckenabschnitte orientiert. Durch diese Vorauswahl ergaben sich für ein paar wenige Kategorien (z.B. beschädigte Bundesstraßen mit Betonbelag) sehr wenige Bilder. In solchen Fällen wurde ein größerer Prozentteil oder alle Bilder pro Bundesland geladen, sodass die Testdaten für diese Kategorien nicht zu gering ausfallen.

9.1.2 Testdaten aus Österreich

Die Auswahl der Testdaten aus Österreich erfolgte direkt aus dem Datenbestand des AIT. Um eine möglichst breite und diverse Auswertung durchzuführen, wurden verschiedene Befahrungssequenzen selektiert, die in Bezug auf Zustand und Oberflächenart auf jeweils möglichst kurzen Strecken eine sehr hohe Diversität aufweisen. Insgesamt wurden 20 Abschnitte auf Rampenanlagen, A+S sowie Landesstraßen für die gegenständliche Testreihe ausgewählt. Diese enthalten sowohl Beton- als auch Asphaltabschnitte und stammen alle aus Messkampagnen des Jahres 2021. Über alle ausgewählten Sequenzen wurden für die Testdaten aus Österreich 100 Kilometer Asphaltbereiche und 25 Kilometer Betonbereiche ausgewählt.

9.2 Auswertung nach ZEB-Standard

Da die Daten für Deutschland über den IT-ZEB Server bildweise bezogen werden können, konnte hier gezielt zwischen den Oberflächenarten Asphalt und Beton unterschieden werden. Entsprechend erfolgt auch die Auswertung getrennt. Für die Auswertung wurde immer die komplette Verarbeitungspipeline nach dem deutschen Standard durchlaufen. So wurde

nach der eigentlichen Detektion, die Nachverarbeitung und anschließend die länderspezifische Konvertierung durchlaufen. Der eigentliche Vergleich erfolgt dann nur noch auf dem ZEB-konformen Bewertungsraster (10x3 Grid pro Bild auf Asphaltabschnitten, plattenweise auf Betonabschnitten).

9.2.1 Asphalt

Wie bereits im vorigen Abschnitt erwähnt, erfolgte die Auswahl von Daten in 3 Kategorien. Im Folgenden werden jeweils zuerst die schadenfreien Abschnitte betrachtet und anschließend die geschädigten Straßenabschnitte.

Für die Bewertung auf Asphaltoberflächen konnten 229,43 Kilometer von dem IT-ZEB-Server heruntergeladen und automatisiert verarbeitet werden. Für die Auswertung der geschädigten Strecken konnten die Abschnitte aus Brandenburg leider keine Verwendung finden⁸, so dass auf Asphalt insgesamt 202,18 km von 2879 verschiedenen und räumlich getrennten Straßenabschnitten genutzt werden konnten.

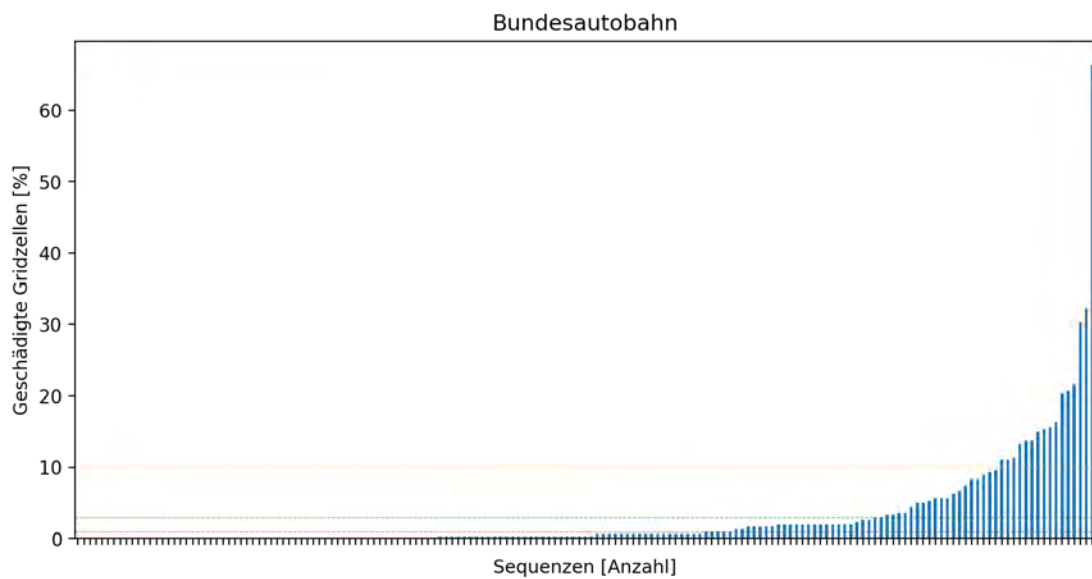
Schadenfreie Streckenabschnitte

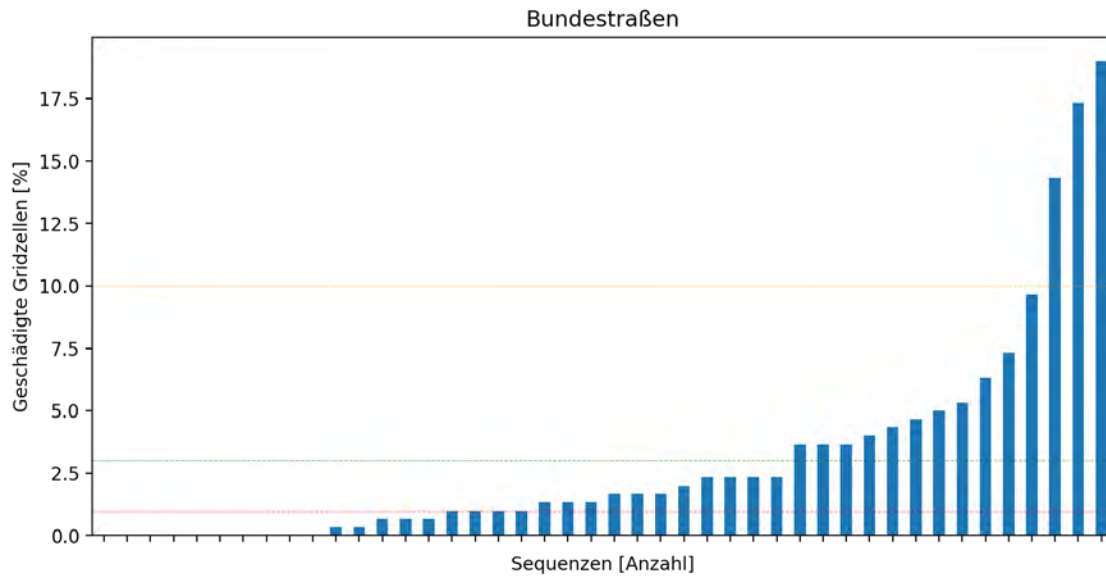
Für die Bewertung der schadenfreien Strecken wurden insgesamt 16,9 Kilometer Autobahn (169 Streckenabschnitte) und 2,4 Kilometer Bundesstraße (44 Streckenabschnitte; teilweise geringere Länge als 100 Meter) ausgewertet. Um zu bewerten, wie gut das automatisierte System diese Strecken auch als schadenfrei erkennt, ist in Abbildung 48 die Verteilung der als schadhaft detektierten Gridzellen über alle Sequenzen (100-Meter-Abschnitte) dargestellt.

Bei den Autobahnabschnitten werden demzufolge 64,5 % der Sequenzen (109) mit weniger als 1% schadhafter Gridzellen detektiert. Bei 79,3 % der Sequenzen (134) werden weniger als 3% der Gridzellen als schadhaft ausgewiesen. Wenn die Toleranz auf 10% Abweichung zur menschlichen Bewertung erhöht wird, fallen 90,5 % der Sequenzen (153) unter diese Abweichungsschwelle. Bei den Bundesstraßen liegen die Anzahl an Sequenzen für die ersten beiden Schwellwerte niedriger. So werden nur bei 43,2 % (19) der Sequenzen weniger als 1% und bei 68,2% (30) der Sequenzen weniger als 3% der Gridzellen als schadhaft markiert. Auf ähnlichem Niveau bewegt sich dann wieder der 10% Schwellwert, so dass dafür 93% der Sequenzen (41) die Bedingung erfüllen.

⁸ Für Brandenburg konnten die Oberflächenbilder nicht den Georohdaten zugeordnet werden, da die originalen Bildnahmen nicht in den Georohdaten zugeordnet werden konnten. Ob ein Fehler bei dem Download der Georohdaten entstanden ist, oder ob der Fehler an anderer Stelle zu suchen ist, konnte bis zum Projektende nicht geklärt werden.

Um zu analysieren, auf welchen Umstand die als geschädigt markierten Gridzellen zurückzuführen sind, wurden die Sequenzen auszugswise noch einmal visuell gesichtet. Dabei konnten verschiedene Fehlerquellen gefunden werden, die zur Aktivierung von Gridzellen geführt haben (Abbildung 49). In einigen Fällen sind solche Detektionen auf Ausprägungen von Klassen zurückzuführen, die in den Trainingsdaten nicht oder nur unzureichend vertreten waren. So wurden z.B. abgetragene Markierungen in einigen Fällen als Aufgelegte Flickstellen detektiert (Abbildung 49 (d)) oder ein feine Laufnase von Markierungsfarbe als Riss (Abbildung 49 (b)). Auch kam es vor, dass Nahtabschnitte fehlerhaft als Nahtrisse detektiert wurden, auch wenn diese eher eine intakte Naht darstellen (Abbildung 49 (a)). In vielen Fällen, wurden jedoch insbesondere Risse sehr gut detektiert, so dass einige der "Fehldetektion" auch auf tatsächliche Schäden zurückgeführt werden können, die bei der menschlichen Bewertung übersehen wurden (Abbildung 49 (c)).





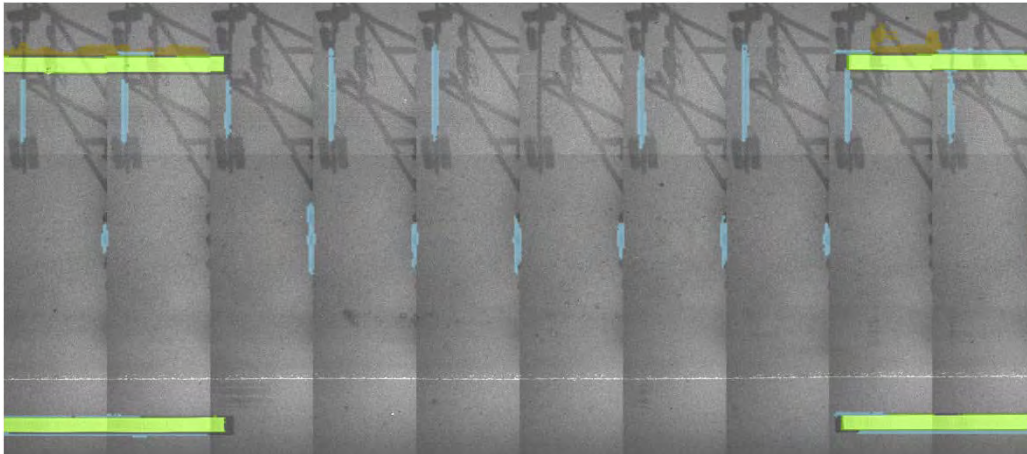


Abbildung 50: Systematische Fehler bei unbekanntem Aufnahmesystem - Die starken Schattenwürfe durch starke Sonneneinstrahlung und ein unbekanntes Aufnahmesystem führen zur systematischen Fehldetektion von Flickstellen an Bildnahtstellen und bei den Schatten des Kameraträgers.

Diese Sequenzen können zu einer sehr großen Abweichung zu der menschlichen Auswertung führen, da hier großflächig Gridzellen als schadhaft markiert werden. Im Projekt wurden bereits Lösungsansätze für diese systematischen Probleme diskutiert. Es wäre beispielsweise möglich, über der Flickstellendetektion eine Fast-Fourier-Analyse durchzuführen und so prüfen, ob die Detektionen die gleiche Frequenz aufweisen, wie die Anzahl der zu einem 10-Meter-Bild gestitchten Bilder. Eine Analyse dieses Ansatzes konnte im Rahmen des ASFaLT-Projekts jedoch nicht mehr durchgeführt werden.

Geschädigte Streckenabschnitte

Für die Analyse von Streckenabschnitten, welche bei der menschlichen Auswertung als fehlerhaft bewertet wurden, konnten insgesamt 73,68 Streckenkilometer aufgeteilt auf 1116 Streckenabschnitte analysiert werden. Die große Anzahl an Sequenzen und die hohe Streckenlänge erschwert die qualitative Analyse im Rahmen des im Projekt vorgesehenen Arbeitsrahmens. Daher wurde für diese Streckenabschnitte nur die Korrelation zwischen den von der KI ermittelten Schadenswerten und der manuellen Auswertung berechnet. Dazu werden die Schadenswerte pro Bild aufsummiert und einander gegenübergestellt.

Die Ergebnisse dieser Analyse sind in Tabelle 16 zusammengefasst.

Tabelle 16: Oberflächenart Asphalt. Korrelationswerte für den Vergleich der manuellen Auswertung mit den Ergebnissen der KI. In den hinteren Spalten ist die Anzahl an Detektionen für die manuelle Bewertung (H) und für die KI dargestellt.

| Typ | Note | km | #Ab. | Korrelationswerte R | | | | #RISS | | #ONA | | #AUS | | #FLI | |
|-----|------|------|------|---------------------|-----|-----|-----|-------|-------|------|------|------|-----|-------|-------|
| | | | | RISS | ONA | AUS | FLI | H | KI | H | KI | H | KI | H | KI |
| B | -3 | 53,8 | 918 | .48 | .35 | .09 | .22 | 3179 | 6354 | 271 | 960 | 83 | 63 | 1955 | 2115 |
| B | -5 | 79,9 | 1252 | .70 | .35 | .16 | .16 | 20279 | 19896 | 1139 | 1496 | 226 | 180 | 1803 | 5163 |
| BAB | -3 | 19,9 | 203 | .48 | .06 | .10 | .37 | 10501 | 25248 | 2977 | 2407 | 163 | 129 | 11459 | 5842 |
| BAB | -5 | 29,3 | 293 | .67 | .62 | .04 | .17 | 58533 | 74261 | 4169 | 4846 | 507 | 609 | 17878 | 12694 |

Bei genauerer Betrachtung der Klasse RISS fällt die Korrelation zunächst geringer aus als erwartet, da diese Klasse auf den gelabelten Daten in der Regel sehr gut detektiert werden konnte. Ein Blick auf die Anzahl der Detektionen verdeutlicht, dass die KI für diese Klasse vermutlich noch zu sensitiv reagiert. Dies könnte beispielsweise daran liegen, dass die Mindestlänge, ab der eine Detektion als Riss stattfindet, bei einer menschlichen Bewertung höher liegt, als diese aktuelle für die KI der Fall ist.

Bei dem Schadensmerkmal ONA hingegen liegt die Anzahl der Detektionen bei 3 von 4 Sequenztypen dicht beieinander. Dennoch ist die Korrelation teilweise gering. Aussagen für dieses Verhalten bedürfen auf jeden Fall einer weiteren Analyse. Da die Anzahl in der Regel recht gut übereinstimmt, stellt sich beispielsweise die Frage, ob es bei dieser Klasse Probleme bei der Lokalisation gab, die zu einer systematischen Abweichung bei der Kennzeichnung der Gridzelle geführt haben. Gleiches gilt für die Schadenart Ausbruch, die nur eine sehr geringe Korrelation aufweist, wobei die Anzahl der Detektionen grundsätzlich optimistisch stimmt. Bei Klassen, die nur wenige Detektionen aufweisen, kann beispielsweise auch eine andere Einpassung des Gridrasters bereits eine große Abweichung in der Korrelationswerten verursachen. Würde beispielsweise durch eine Verschiebung des Grids immer die Nachbarzelle aktiviert, würde das die Korrelation deutlich verschlechtern. Um den Einfluss dieses Fehlers zu reduzieren, könnte ein Vergleich auf Basis von 100-Meter-Zustandswerten erfolgen. Dieser Vergleich konnte im Rahmen des Projekts jedoch nicht mehr durchgeführt werden.

Die Korrelationswerte für die Detektion von Flickstellen werden auf jeden Fall durch zwei systematische Probleme negativ beeinflusst. Zuerst stellt der Flächenausgleich hier ein großes Problem dar, da dieser von Menschen nur „in etwa“ durchgeführt werden kann, was bereits zu großen Abweichungen in der Anzahl der markierten Zellen führen kann⁹. Ein weiteres Problem ergibt sich aktuell noch bei Flickstellen, die sich über die gesamte Fahrbahnbreite erstrecken. In diesem Fall würde durch die KI nur der Übergangsbereich als Flickstelle markiert werden, wodurch es zu großen Abweichungen in der Fläche kommen kann.

9.2.2 Beton

Die Länge der analysierten Strecken mit der Oberflächenart Beton fällt deutlich geringer aus als die Länge mit der Oberflächenart Asphalt. Dies ist zum einen dem Umstand geschuldet, dass die Oberflächenart generell weniger vertreten ist und außerdem sehr unter dem Ausfall der Bewertungen für Brandenburg leidet. Zum anderen waren die Betonbilder deutlich mehr von der Problematik der fehlerhaften Downloads betroffen. Infolgedessen konnten auf der Oberflächenart Beton lediglich 34,22 Streckenkilometer verteilt auf 89 Streckenabschnitte ausgewertet werden.

Schadenfreie Streckenabschnitte

Analog zu der Auswertung der Asphaltabschnitte ist auch für die Betonabschnitte die Verteilung der als schadhaft detektierten Gridzellen über alle Sequenzen erstellt worden (Abbildung 51).

Es zeigt sich, dass der auf den Betondaten trainierte Detektor auf schadenfreien Strecken deutlich mehr Detektionen von Schäden produziert. So fallen auf der Bundesstraße nur 3% (1 Sequenz), 11% (3 Sequenzen), und 33% (9 Sequenzen) unter die Schwellen von 1%, 3% und 10%. Etwas besser stellt sich das Bild für die Bundesautobahnen dar, wo entsprechend 28% (24 Sequenzen), 43% (37 Sequenzen) und 59% (51 Sequenzen) unter die Schwellenwerte fallen. Die hohe Anzahl an falsch-positiv Detektionen lässt sich vor allem auf die ungünstigere Datenlage für die Oberflächenart Beton zurückführen.

⁹ Kurz vor Ende des Projekts stellte sich außerdem heraus, dass der Flächenausgleich durch die KI anders umgesetzt wurde, als dies in der Praxis der händischen Bewertung der Fall ist. Während bei der manuellen Bewertung nur ein Flächenausgleich in horizontaler Richtung erfolgt, so dass die Länge der Fläche auf jeden Fall immer komplett gelabelt wird, werden durch die KI immer nur so viele Zellen markiert, dass sie der tatsächlichen Fläche der Flickstelle entsprechen.

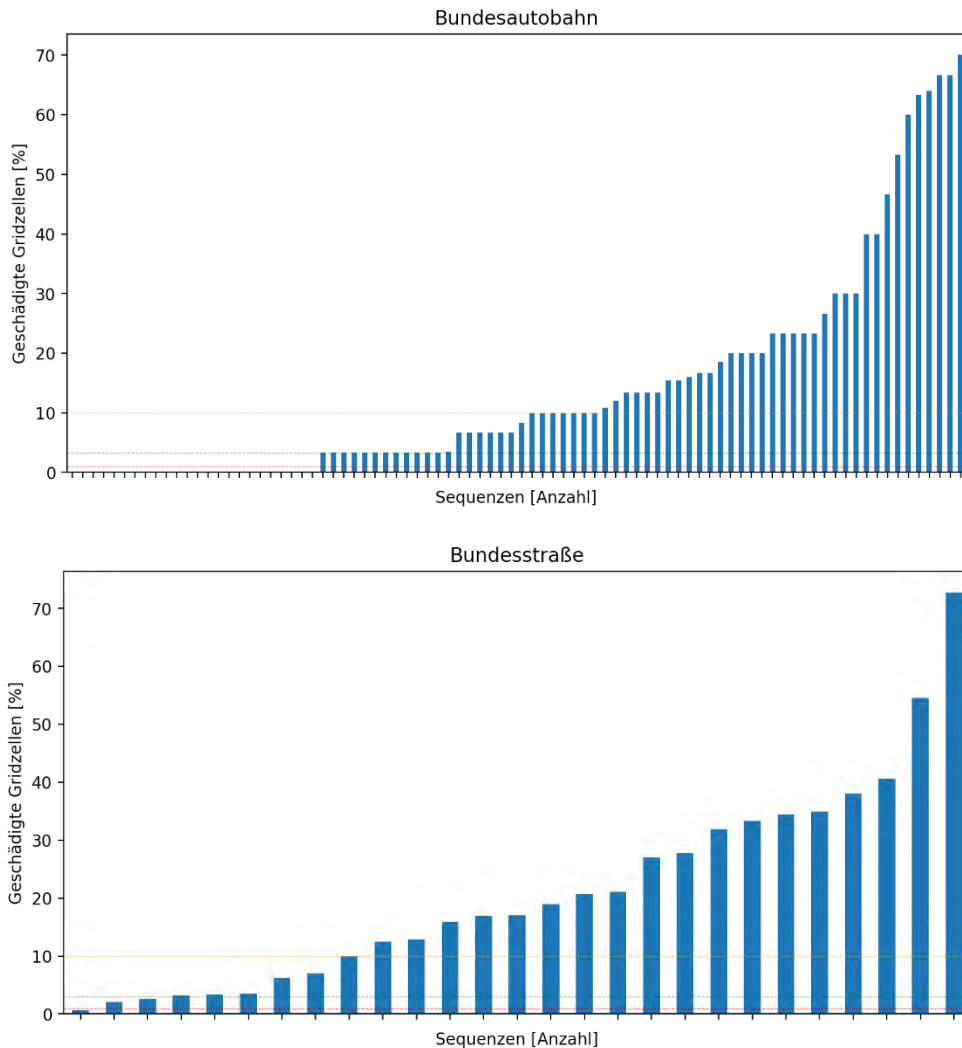


Abbildung 51: Automatisierte Auswertung von schadenfreien Betonstreckenabschnitten - die Grafiken zeigen für die Streckenabschnitte auf Bundesautobahnen und Bundesstraßen die jeweils als geschädigt ausgegebene Gridzellen pro Abschnitt in Prozent. Die gestrichelten Linien in rot, grün und gelb geben als Orientierung 1%, 3% und 10% der Gesamtanzahl an Gridzellen pro Sequenz an.

Bereits in einer frühen Version des Detektors hatte sich gezeigt, dass dieser Probleme mit den unterschiedlichen Betontypen zu haben scheint. Manche Betonoberflächen weisen grundsätzlich eine „risshafte“ Optik auf, während auf anderen Betonoberflächen bereits sehr feine Risse durch menschliche Auswerter bewertet werden. Dies ist ein Grundproblem, mit denen der aktuelle Detektor auf Grundlage der aktuellen Datenbasis noch nicht richtig umgehen kann.

Geschädigte Streckenabschnitte

Die berechneten Korrelationen auf den Ergebnissen auf den geschädigten Streckenabschnitten, welche ebenfalls analog zu der Asphaltauswertung durchgeführt wurde, weisen eine sehr große Streubreite innerhalb der einzelnen Klassen auf (Tabelle 17).

Tabelle 17: Oberflächenart Beton. Korrelationswerte für den Vergleich der manuellen Auswertung mit den Ergebnissen der KI.

Korrelationswerte R

| Typ | Note | km | # Ab. | LQR | EAB | KAS | NTR | BTE |
|------------|------|------|-------|-------|------|------|-------|------|
| B | ~3 | 2,22 | 22 | 0.44 | 0.31 | 0.44 | 0.18 | 0.46 |
| B | ~5 | 3,83 | 38 | -0.14 | 0.17 | 0.32 | 0.04 | 0.49 |
| BAB | ~3 | 0,78 | 11 | 0.34 | - | 0.08 | 0.18 | 0.05 |
| BAB | ~5 | 1,53 | 18 | 0.42 | 0.53 | 0.10 | -0.03 | 0.18 |

Eine Auswertung gestaltet sich auf Grundlage dieser Werte daher als schwierig. In Anbetracht der geringen Datenlage bei den Testdaten und der nicht ausreichenden Datenlage bei den Trainingsdaten scheint eine Interpretation daher wenig zielführend zu sein. Im Rahmen von über das Projekt hinausgehenden Arbeiten müssten daher auf Grundlage einer qualitativen Untersuchung zuerst weitere systematische Probleme mit dem Detektor bei der Anwendung auf weiteren ZEB-Daten analysiert werden. In der Domäne Beton würden sehr wahrscheinlich weitere Trainingsdaten am meisten helfen, um die Detektion deutlich zu verbessern.

9.2.3 Fazit

Die erzielten Ergebnisse bei der ZEB-Auswertung zeigen ein recht ambivalentes Bild. Während beispielsweise die Rissdetektion und die Detektion von schadenfreien Strecken auf Asphaltoberflächen bereits eine recht gute Qualität aufweisen, erreicht die Detektion von Flächenklassen und auch die Detektion auf Beton aktuell noch nicht das für die Automatisierung angestrebte Niveau.

Neben einer Erweiterung der Trainingsdaten bedarf es vor allem einer ausgiebigen Analyse der Fehlerquellen, um auch den Vergleich zwischen der menschlichen Bewertung und der KI noch zu verbessern. Aktuell stellt schon allein der Vergleich auf Grundlage der ZEB-Raster eine große Unsicherheitsquelle dar (z.B. unbekannte Position des Grids, Flächenausgleich). Um dieses Problem zu reduzieren, könnte in möglichen Folgeprojekten auch die Bewertung auf 100-Meter-Werten berechnet werden. Bei dem Vergleich zwischen menschlicher Bewertung und automatisierter Bewertung auf Grundlage von Segmentierungen hatte sich außerdem gezeigt, dass auch die menschliche Bewertung eine sehr große Streubreite aufweist. Auch dieser Faktor müsste daher noch genauer analysiert werden, um die Ergebnisse vergleichen zu können.

9.3 Auswertung nach RVS-Standard

Im Fall von Österreich ist hier die RVS 13.01.16 maßgeblich. Diese beschreibt die Berechnung der Zustandsgrößen für Risse und Oberflächenschäden auf den Belägen Asphalt und Beton anhand der Summe der gewichteten geschädigten Flächen im Verhältnis zur Bezugsfläche. Das Ergebnis stellt den Prozentwert der geschädigten Fläche dar. Desweiteren erfolgte in diesem Fall die Endauswertung mit 50 m-Mittelwerten, da dies in Österreich sowohl für das A+S-Netz, als auch das Landesstraßennetz die maßgebliche Abschnittslänge darstellt.

Für die Auswertung wurde die unter 9.1 angegebene Streckenlänge von 125 Kilometer nach RVS automatisiert bewertet. Da diese Tests in der Endphase des Projektes durchgeführt wurden, war es aus zeitlichen und ressourcentechnischen Gründen nicht mehr möglich, in diesem Bericht eine umfassende Auswertung bzw. Analyse der erhaltenen Ergebnisse vorzunehmen. Der Fokus wäre in diesem Fall auf der genauen Untersuchung der Ursachen allfälliger Abweichungen zwischen menschlicher und KI-Auswertung sowie deren Reduzierung gelegen. Exemplarisch wird im Folgenden jedoch die visualisierte Gegenüberstellung der menschlichen und der KI-Auswertung gemäß RVS 13.01.16 auf einer Landesstraße, zwei Autobahnabschnitten sowie einer Rampenanlage angeführt.

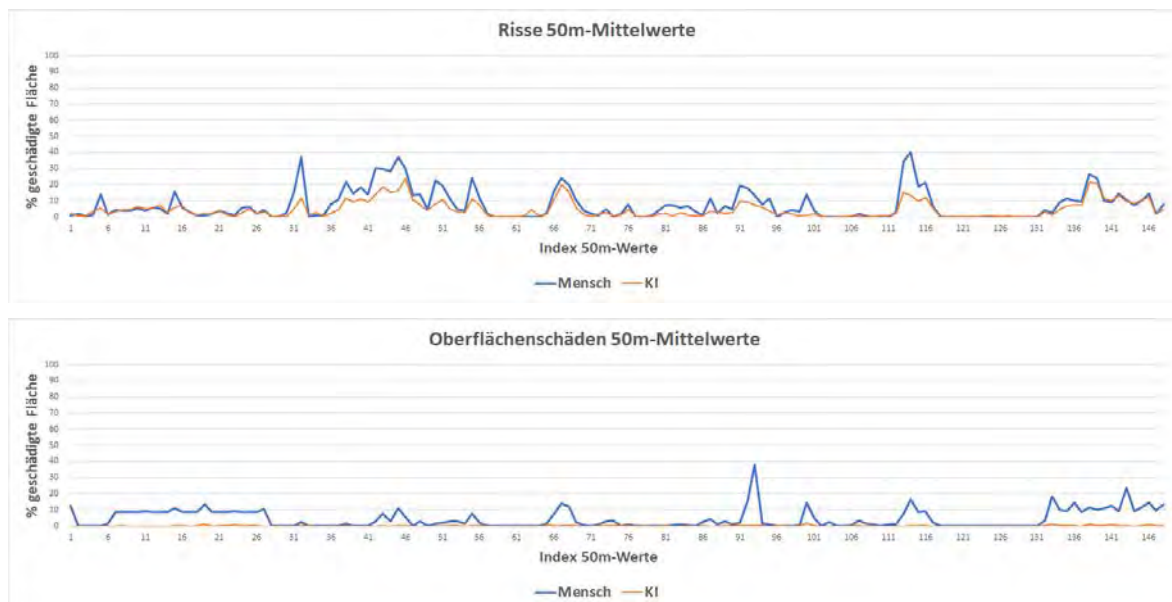


Abbildung 52: Landesstraße – Vergleich Risse und Oberflächenschäden, Mensch-KI

Die Verläufe der Auswertung der Risse der dargestellten *Landesstraße* in Abbildung 52 sind bis auf einzelne Spitzen vergleichbar. Die menschliche Auswertung der Oberflächenschäden zeigt grösstenteils Bereiche im - für Landesstraßen - niedrigen Schadensbereich von unter 10%. Die KI hat diese nicht als solche interpretiert. Abbildung 53 zeigt den Vergleich Mensch-KI nach Überführung in Zustandsklassen gem. RVS 13.01.16.

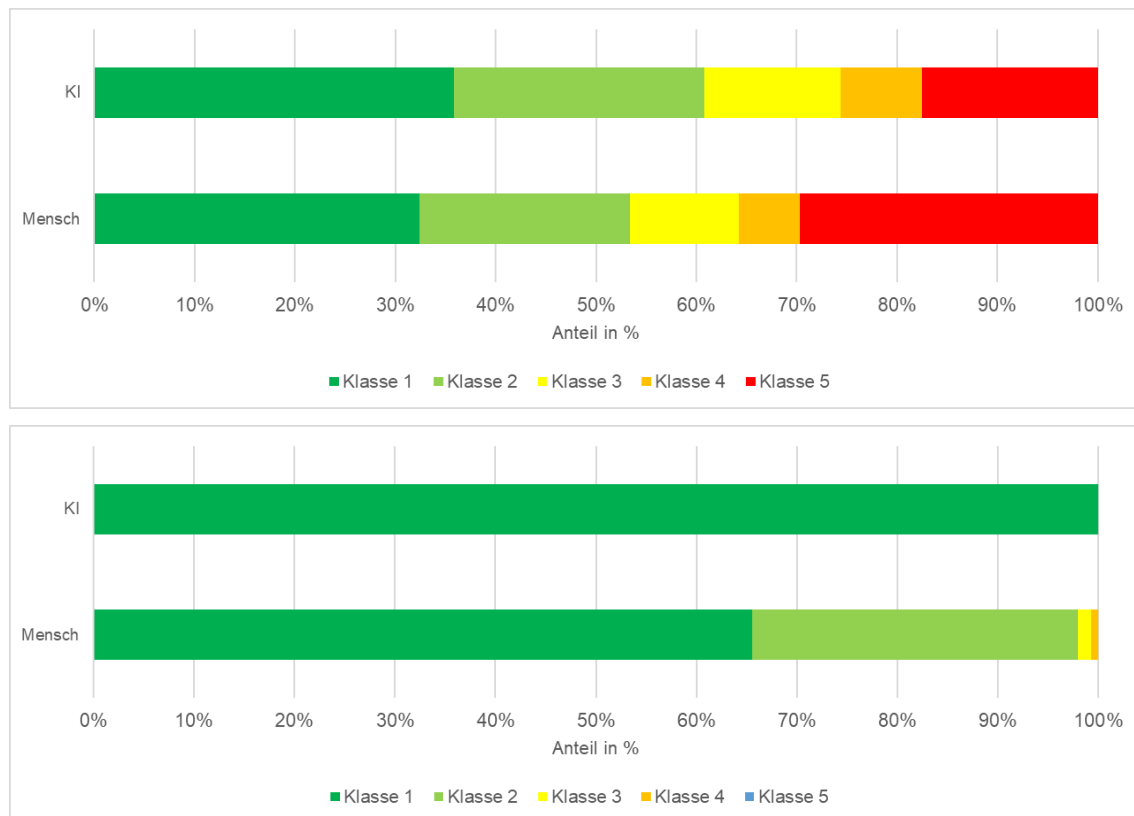


Abbildung 53: Vergleich der Anteile der Zustandsklassen gemäß RVS 13.01.16 für Risse (% geschädigte Fläche, oben) und Oberflächenschäden (% geschädigte Fläche, unten) für die Auswertung durch KI und Mensch auf der Landesstraße

Bei den Verläufen der Risse des dargestellten *Autobahnabschnittes 1* (siehe Abbildung 54) gibt es einen grossen Bereich, in dem die menschliche Auswertung keine Schäden erkannt hat, die KI dagegen Schäden im niedrigen einstelligen Bereich. Danach folgt ein Abschnitt, auf dem sowohl die Risse in der KI-, als auch der menschlichen Auswertung deutlich erhöht sind und grundsätzlich vergleichbar erscheinen. Beide Auswertungen hinsichtlich der Oberflächenschäden ergeben im vorderen Bereich defacto keine Schäden. Im hinteren Bereich gibt es bei der KI nur am Beginn einen deckungsgleichen Ausschlag, während danach nur noch bei der menschlichen Auswertung Spitzen auftreten. Abbildung 55 zeigt den Vergleich Mensch-KI nach Überführung in Zustandsklassen gem. RVS 13.01.16.

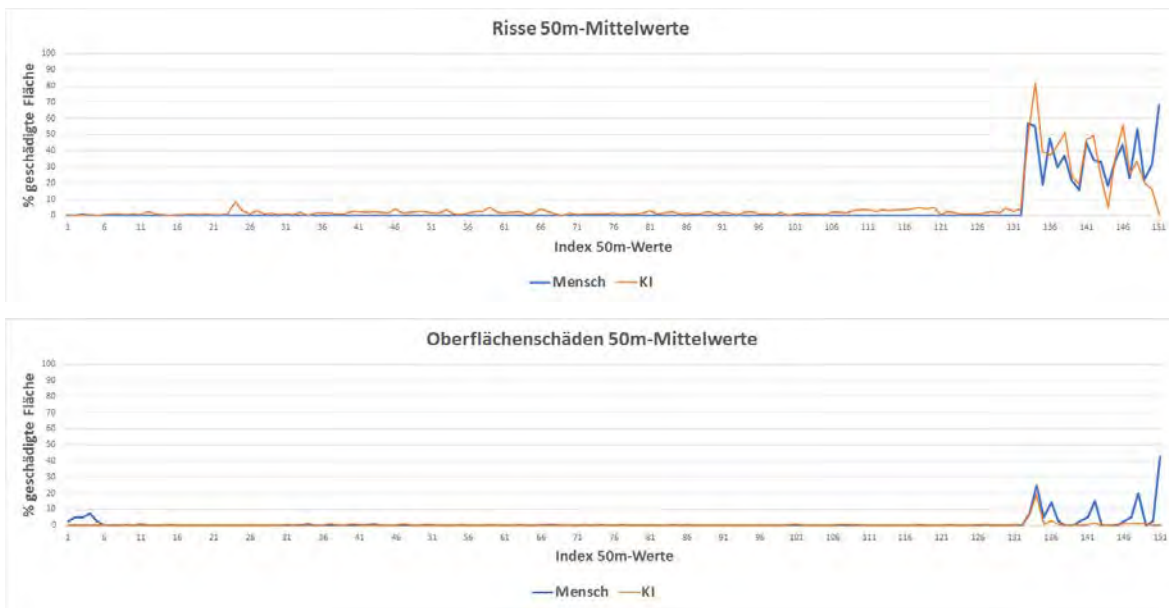


Abbildung 54: Autobahnabschnitt 1 – Vergleich Risse und Oberflächenschäden, Mensch-KI

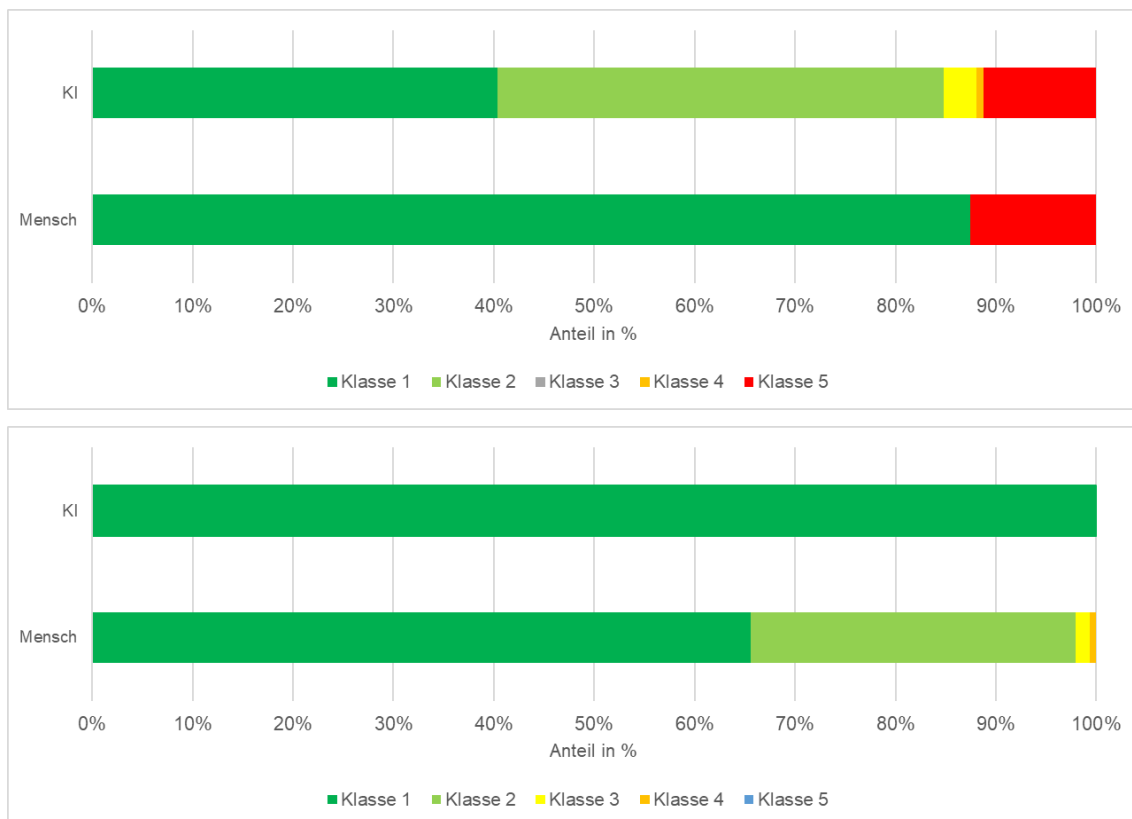


Abbildung 55: Vergleich der Anteile der Zustandsklassen gemäß RVS 13.01.16 für Risse (% geschädigte Fläche, oben) und Oberflächenschäden (% geschädigte Fläche, unten) für die Auswertung durch KI und Mensch auf dem Autobahnabschnitt 1

In Abbildung 56 – *Autobahnabschnitt 2* – befinden sich sowohl bei den Rissen als auch bei den Oberflächenschäden die Werte der menschlichen und der KI-Auswertung überwiegend im niedrigen einstelligen Bereich. Ausnahmen sind einzelne Spitzen, die bei den Rissen die KI betrifft, bei den Oberflächenschäden hingegen die menschliche Auswertung. Abbildung 57 Abbildung 53 zeigt den Vergleich Mensch-KI nach Überführung in Zustandsklassen gem. RVS 13.01.16.

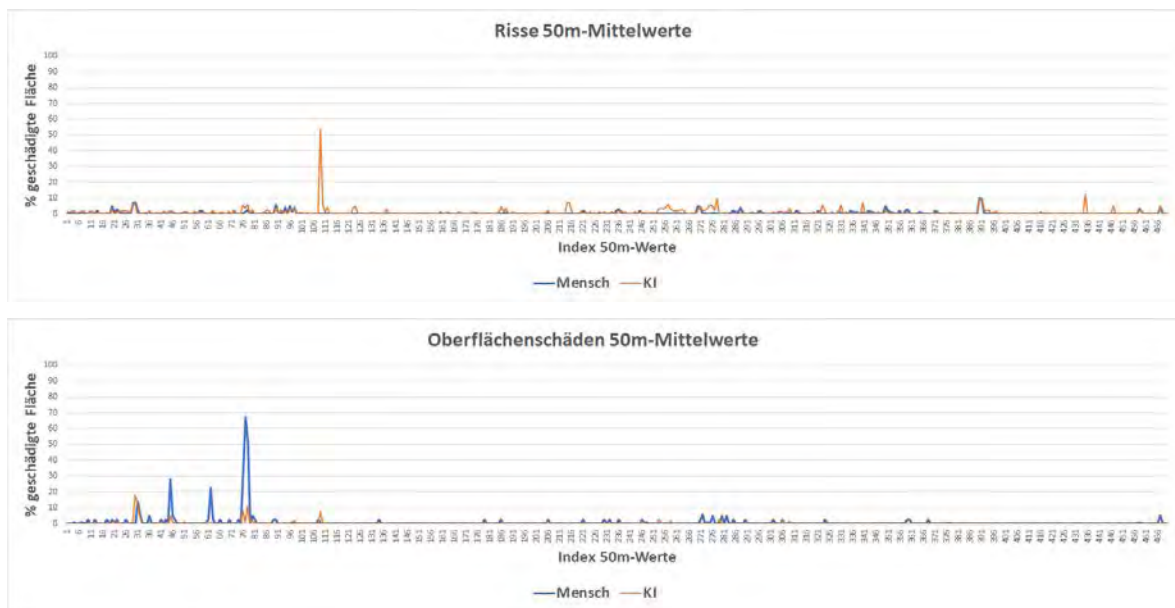
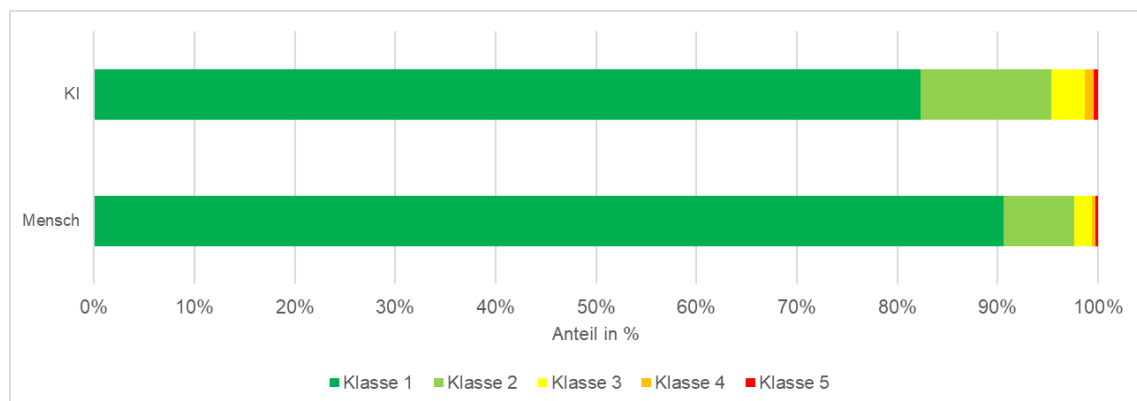


Abbildung 56: *Autobahnabschnitt 2* – Vergleich Risse und Oberflächenschäden, Mensch-KI



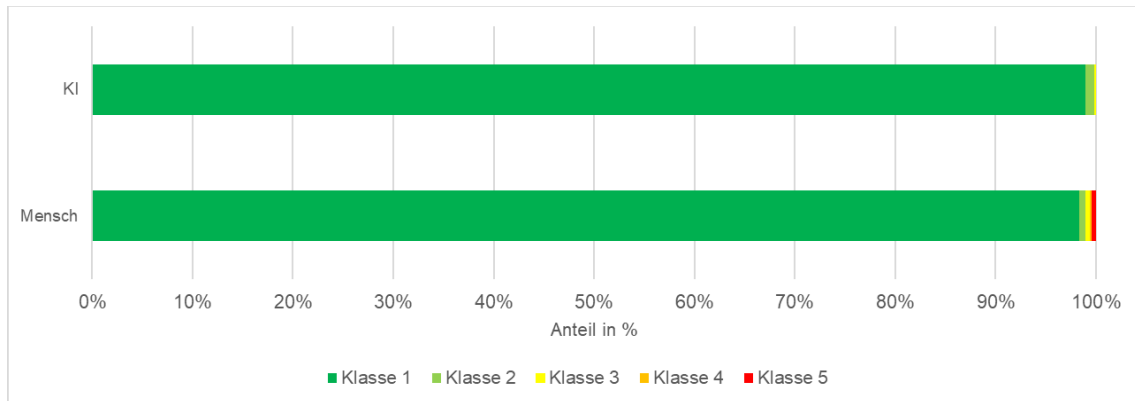


Abbildung 57: Vergleich der Anteile der Zustandsklassen gemäß RVS 13.01.16 für Risse (% geschädigte Fläche, oben) und Oberflächenschäden (% geschädigte Fläche, unten) für die Auswertung durch KI und Mensch auf dem Autobahnabschnitt 2

Der Verlauf der Risse in Abbildung 58 – *Rampenanlage* – ist für beide Auswertarten vergleichbar. Bei den Oberflächenschäden hat die KI-Auswertung am Beginn einen Bereich großer Abweichung im Vergleich zur menschlichen Auswertung danach sind die beiden Auswertungen gut vergleichbar. Abbildung 59 zeigt den Vergleich Mensch-KI nach Überführung in Zustandsklassen gem. RVS 13.01.16.

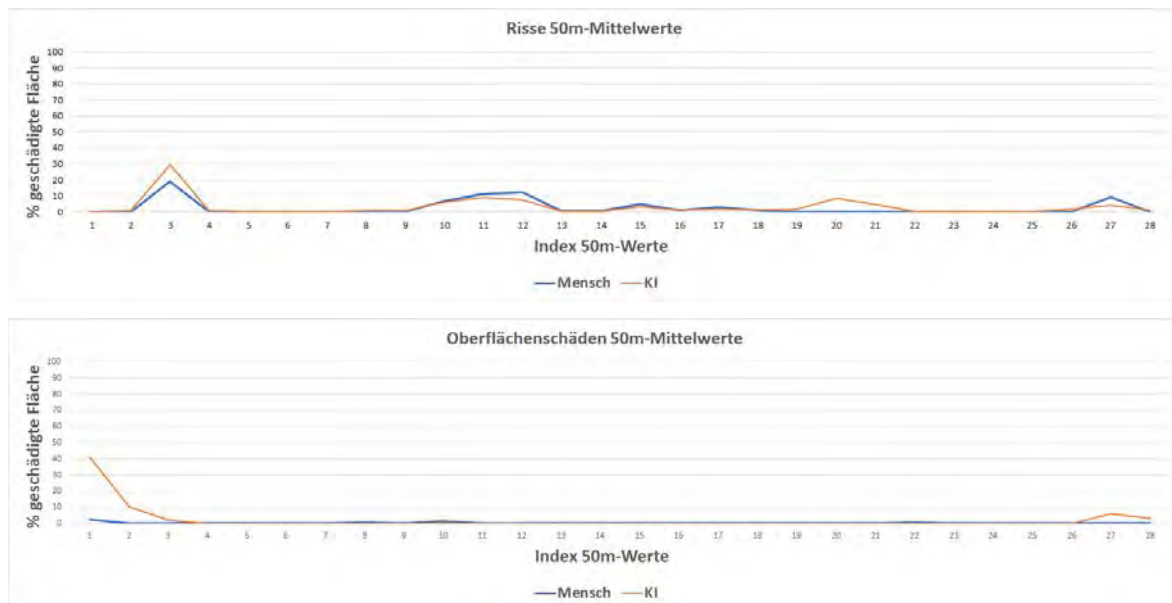


Abbildung 58: Rampenanlage – Vergleich Risse und Oberflächenschäden, Mensch-KI

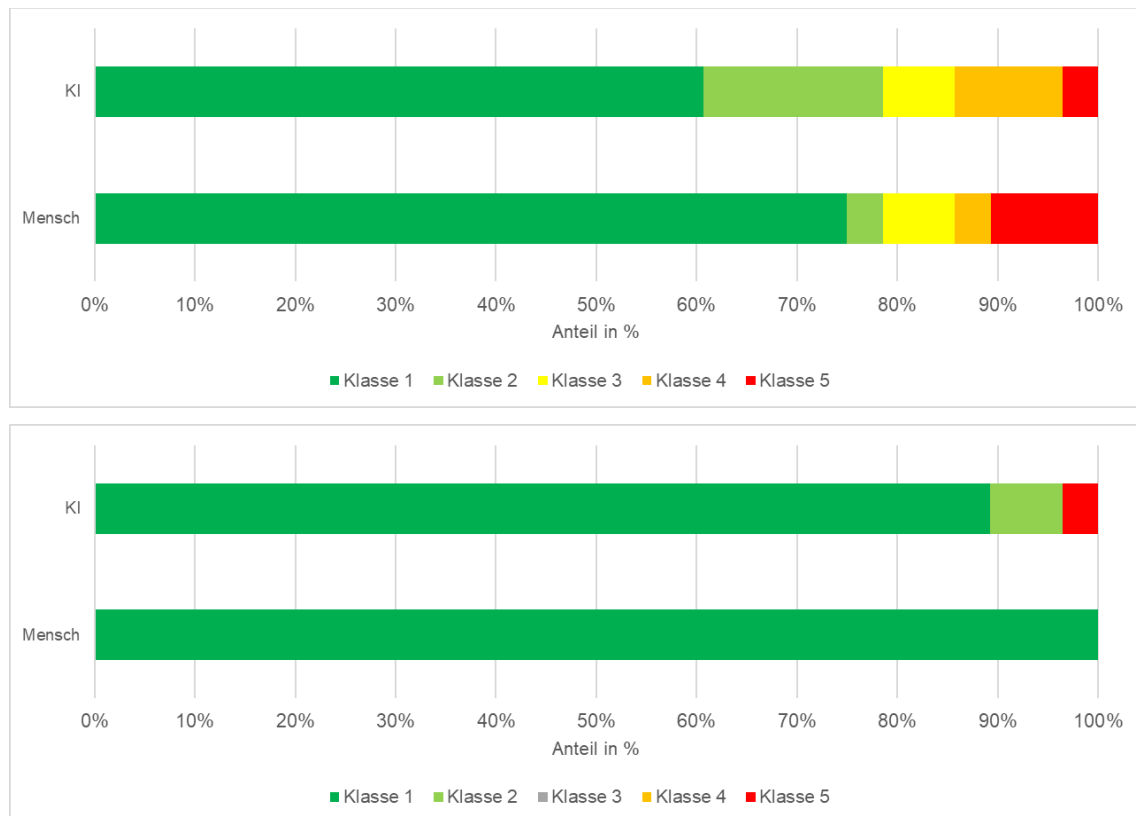


Abbildung 59: Vergleich der Anteile der Zustandsklassen gemäß RVS 13.01.16 für Risse (% geschädigte Fläche, oben) und Oberflächenschäden (% geschädigte Fläche, unten) für die Auswertung durch KI und Mensch auf der Rampeanlage

9.3.1 Fazit

Insbesondere für die Klasse der Risse konnte in der Gegenüberstellung eine große Deckung zwischen der menschlichen Bewertung und der Bewertung durch das KI-System festgestellt werden. Bei den Oberflächenschäden ergeben sich hingegen noch größere Discrepanzen, die einer gezielten Analyse auf Bildebene bedürfen. Für die Bewertung nach RVS spielen hierbei auch eine Reihe an Schwellwerten eine Rolle, die die Bewertung maßgeblich beeinflussen. Da beispielsweise die Schäden mit einer kleinsten Einheit von 1 m bzw. 1 m² bewertet werden, ist zu analysieren, ab welchem Schadensgrad der Mensch eine Bewertung vornimmt. Dementsprechend sollten die Parameter der KI entsprechend adaptiert werden.

Die genaue Analyse der Sequenzen und die sich daran anschließende Adaption der Parameter bietet Raum für weitere Untersuchungen im Nachgang des Projekts.

9.4 Laufzeit

Die Laufzeit der Detektion hängt stark von der eingesetzten Hardware und hierbei insbesondere von der eingesetzten Grafikkarte ab. Auf einer NVIDIA GeForce 2080 Ti benötigt beispielsweise die reine Detektion für ein 10-Meter-Bild (mit 450 Pixel pro Meter Auflösung) etwa 3 Sekunden. Zuzüglich der Nachverarbeitung ist daher die Verarbeitung in etwa 5 Sekunden möglich. Auf diese Weise können mit einer Grafikkarte in der Stunde etwa 7 Kilometer Straße bewertet werden. Bei Hinzunahme der Modellunsicherheit verlängert sich die Verarbeitungszeit auf etwa 11 Sekunden pro Bild, so dass dann etwas mehr als 3 Kilometer pro Stunde bewertet werden können.

Durch Parallelisierung über mehrere Grafikkarten lässt sich die Auswertungsgeschwindigkeit jedoch auf einfache Weise deutlich erhöhen.

9.5 Fazit

Unter bestimmten Rahmenbedingungen zeigt sich, insbesondere für die Detektion von Rissklassen auf Asphaltstraßen, bereits die Praxistauglichkeit des entwickelten Systems. So konnte beispielsweise in Deutschland die recht gute Detektion von schadenfreien Abschnitten und auch in Österreich der korrelierte Verlauf der Risschäden mit der menschlichen Bewertung gezeigt werden.

Unterschiedliche Kamerasysteme, extreme Beleuchtungen und Flächenklassen stellen aktuelle jedoch noch Unsicherheitsfaktoren dar, die noch genauer untersucht und auch optimiert werden müssen.

Grundsätzlich weist der Vergleich mit der menschlichen Bewertung auf dem ZEB-Raster grundlegende Probleme auf, die die Interpretation auf quantitativer Ebene erschweren. Eine genauere qualitative Untersuchung sollte sich daher an das Projekt ASFaLT anschließen, um beispielsweise auch zu klären, wie die Ergebnisse in das Verhältnis zu zwei menschlichen Auswertergebnissen gesetzt werden können.

Auch bietet das System noch eine Reihe an Parametern, die bisher nur auf den eigentlichen Trainingsdaten optimiert wurden, für die sich aber eine Optimierung auf dem Datenbestand des IT-ZEB-Servers anbietet, um die automatisierte und manuelle Bewertung anzunähern. Nach aktuellem Stand könnte das System gut geeignet sein, um schadenfreie Strecken auf Asphaltoberflächen zu detektieren und auch um genaue Rissdetektion und Risslängen zu extrahieren. Flächenklassen sollten jedoch noch manuell nachkontrolliert werden.

10 Bewertung der Schadensentwicklung

Um die zeitliche Entwicklung von Schäden visuell beurteilen zu können, ist ein dreiphasiges Vorgehen erforderlich. In einem ersten Schritt müssen zunächst Befahrungen von unterschiedlichen Zeitpunkten in räumliche Deckung gebracht werden. In Abschnitt 10.1 wird beschrieben, wie eine solche Synchronisation mithilfe der bei den Befahrungen aufgenommenen GPS-Daten, des Inertialsensors des Aufnahmefahrzeugs und auf Grundlage von stabilen, visuellen Merkmalen der Einsatzumgebung in den aufgenommenen Bildsequenzen umgesetzt wurde. Unter 10.2 wird beschrieben, wie in der sich anschließenden Phase die Synchronisation von Schadstellen gehandhabt werden kann. 10.3 beschreibt, wie im Anschluss eine Analyse der eigentlichen Schadensentwicklung durchgeführt werden kann.

Die Befahrungen erfolgen typischerweise um mehrere Jahre zeitlich versetzt und mit unterschiedlichen Aufnahmesystemen, die über unterschiedliche Kamera- und Beleuchtungssysteme verfügen. Um einige dieser Faktoren auszuschließen, wurden im Rahmen des Projekts Referenzaufnahmen mit identischem Aufnahmesystem durchgeführt (Innerorts - vor und nach Frostwechsel).

10.1 *Räumliche Synchronisation von Straßenbildern*

Unter dem Aspekt der räumlichen Synchronisation von Straßenbildern wurden zwei verschiedene Aufgaben bearbeitet. Zum einen wurde die Synchronisation von mehreren Oberflächenbildern miteinander untersucht. Zum anderen wurde überprüft, ob sich auch Frontaufnahmen und Oberflächenbilder robust synchronisieren lassen, um die Entscheidungsfindung des Netzwerks bei schwer entscheidbaren Fällen verbessern zu können.

10.1.1 **Räumliche Synchronisation von Oberflächenbildern aus zeitlich versetzten Aufnahmen**

Für die Beurteilung der Schadensentwicklung und die dabei notwendige Synchronisation von Schadstellen müssen Befahrungen von unterschiedlichen Zeitpunkten in räumliche Deckung gebracht werden. Nachdem zwei Bildsequenzen mit GPS-Daten aufgenommen wurden, müssen die einzelnen Bilder zueinander ausgerichtet werden, damit eine Analyse der Schadensentwicklung durchgeführt werden kann. In der Masterarbeit [Lembach, 2019] und darauf aufbauenden Arbeiten wurde ein System für die benötigte räumliche Synchronisation von Oberflächenbildern entwickelt. Abbildung 60 zeigt eine Übersicht des entwickelten Systems.

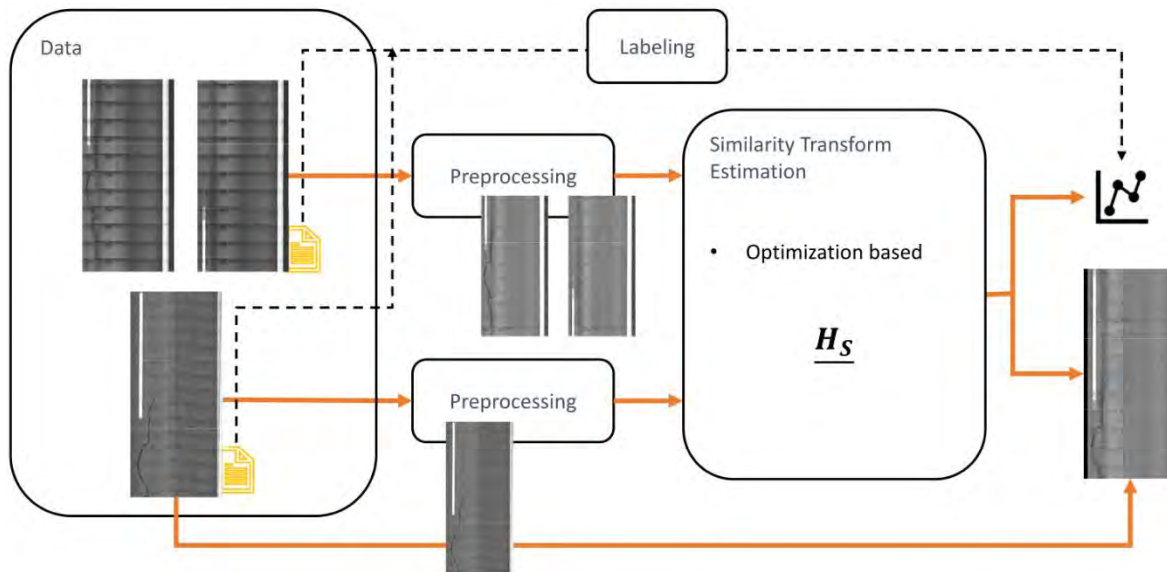


Abbildung 60: Systemübersicht zur räumlichen Synchronisation von Oberflächenbildern. Es wurde ein modulares System zur Ausrichtung von Oberflächenbildern aus verschiedenen Jahren umgesetzt. Das Dateneingangsmodul findet die beiden Einzelbilder, die dem Bezugsbild am nächsten liegen. Nach der Vorverarbeitung wird eine Ähnlichkeitstransformation mittels eines Optimierungsverfahrens geschätzt. Die gefundene Transformation kann mit einer gelabelten Ähnlichkeitstransformation verglichen werden. [Lembach 2019]

Aufbauend auf den Arbeiten in der Masterarbeit [Lembach 2019] wurde der Datensatz mit gelabelten Oberflächenbildern auf insgesamt 244 Bilder erweitert. Die folgende Tabelle gibt eine Übersicht über die Anzahl der Bilder in den jeweiligen Sequenzen und die abgebildeten Szenarien:

Tabelle 18: Überblick der für die Untersuchungen verwendeten Testsequenzen.

| Szenario | Jahre | # Bilder |
|--------------|-------------|----------|
| Autobahn (1) | 2013 - 2017 | 40 |
| Autobahn (2) | 2014 - 2018 | 60 |
| Bundesstraße | 2011 - 2015 | 100 |
| Innerorts | 2018 - 2019 | 44 |
| Gesamt | | 244 |

Für eine räumliche Synchronisation müssen drei Bilder betrachtet werden. Ein Referenzbild aus einer gewählten Aufnahmeserie und zwei Bilder aus der anderen Aufnahmeserie, da die Bilder nicht an exakt gleicher Position aufgenommen werden und sich somit die Bildbereiche des ersten 10-Meter-Bildes über 2 Bilder in der zweiten Sequenz erstrecken. Die zwei weiteren Bilder werden über die Entfernung in den aufgenommenen GPS-Koordinaten oder per Zuordnung über die Dateinamen gefunden und anschließend zusammengeführt,

sodass sich eine doppelt so lange Aufnahme ergibt, in welcher das Referenzbild synchronisiert werden kann.

Hierzu mussten verschiedene Teilaufgaben, wie in Abbildung 60 dargestellt, gelöst werden. So mussten die Bilder zunächst vorverarbeitet werden, um Störfaktoren zu vermindern und die Synchronisation zu vereinfachen. Daraufhin mussten die Bilder mittels eines Optimierungsalgorithmus zueinander ausgerichtet werden. Um die Synchronisation zu bewerten, wurde ein Gütemaß bestimmt, das mit Hilfe der gelabelten Daten berechnet wird. Wenn eine passende Synchronisation gefunden wurde, kann anschließend die zeitliche Veränderung visualisiert werden. Im Folgenden wird auf diese Aspekte näher eingegangen sowie die Ergebnisse zu verschiedenen Varianten vorgestellt.

Vorverarbeitung

Bei Betrachtung der Beispielbilder im linken Teil der Abbildung 60 wird intuitiv klar, weshalb eine Vorverarbeitung der Bilder benötigt wird. In der Masterarbeit [Lembach 2019] wurden drei verschiedene Vorverarbeitungen umgesetzt und in ihren Kombinationen untereinander getestet:

- Rasterausgleich: Das Bild wird bestmöglich von Stitching-Artefakten und den Schatten des Aufnahmearaufbaus befreit.
- Normalisierung: Das Bild wird von seinem Mittelwert befreit und die Standardabweichung auf eins gesetzt.
- Schwellenwertverfahren: Das Bild wird in ein Binärbild umgewandelt.

In folgenden Arbeiten wurde zusätzlich die Normalisierung erweitert, sodass die starken Unterschiede zwischen dunkler Fahrbahn und hellen Markierungsstreifen adaptiv verarbeitet werden können.

Optimierungsansatz und Kostenfunktion

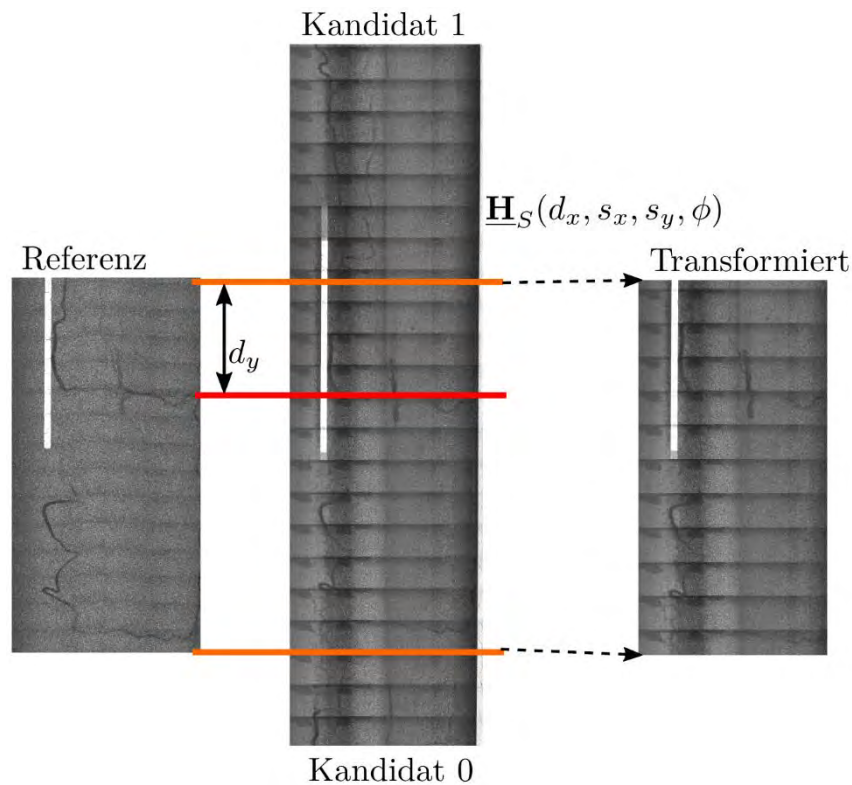


Abbildung 61: Visualisierung des Stitchings und der Ähnlichkeitstransformation. Die Ähnlichkeitstransformation wird realisiert, indem die beiden Kandidaten zuerst bei d_y zusammengesetzt und dann die Skalierung s und die horizontale Verschiebung d_x angewendet werden. [Lembach 2019]

Abbildung 61 beschreibt die gesuchten Parameter des Optimierungsalgorithmus und das Ziel der räumlichen Synchronisation. Ein Referenzbild aus einer Aufnahmeserie soll mit den Bildern einer zweiten Aufnahme räumlich synchronisiert werden. Zu jedem Referenzbild gibt es zwei Kandidaten, auf denen der Straßenabschnitt des Referenzbildes abgebildet ist. Diese werden über die GPS-Koordinaten oder Dateinamen des Referenzbildes gefunden und zusammengeführt.

Zur Synchronisation müssen die Verschiebung in horizontaler d_x und vertikaler d_y Richtung bestimmt werden, sowie die horizontale und vertikale Skalierung s_x, s_y und die Rotation ϕ zwischen den Aufnahmeserien. Die in der Masterarbeit [Lembach 2019] genutzte Ähnlichkeitstransformation (d_x, d_y und $s=s_x=s_y$) wurde in folgenden Untersuchungen durch die oben genannten Parameter erweitert, um so eine höhere Variabilität und Präzision zu ermöglichen. Des Weiteren wurde das in [Lembach 2019] verwendete Ähnlichkeitsmaß, mittlerer quadratischer Pixelfehler (MSE), durch ein informationstheoretisches Maß, Regional Mutual Information (RMI) [Russakoff 2004], ersetzt und ausgiebig untersucht. Abbildung 62 zeigt das deutlich ausgeprägtere Minimum in der Kostenfunktion, das darauf hindeutet,

dass das dort liegende Minimum das Optimum der Transformation ist. Ähnliche Fehlergebirge konnten für einen Großteil der Bilder festgestellt werden.

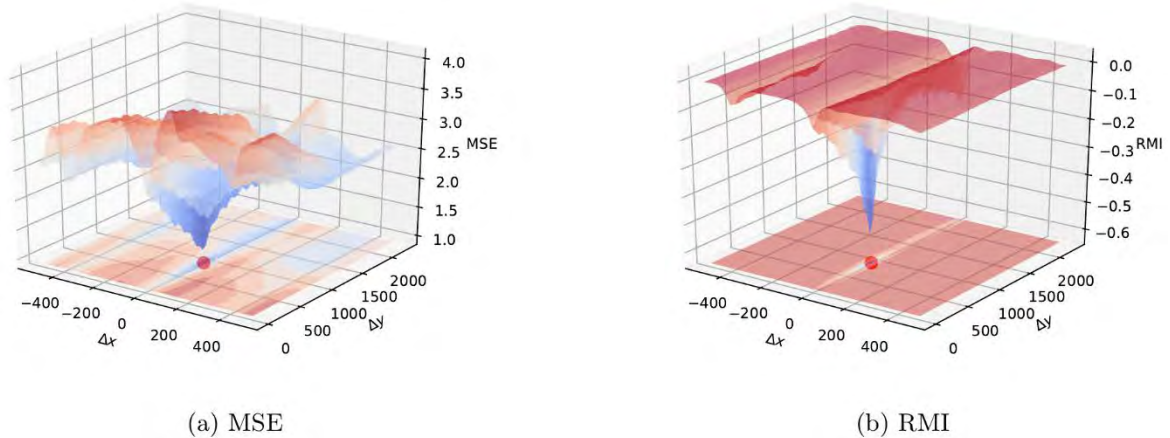


Abbildung 62: Fehlergebirge der Kostenfunktionen bei variabler Translation in x - und y -Richtung. Der Mittlere Quadratische Pixelfehler (a) hat deutlich mehr lokale Minima und ein weniger ausgeprägtes globales Minimum als die Regional Mutual Information (b).

Die Optimierung findet wie in der Masterarbeit [Lembach 2019] durch den Algorithmus *Differential Evolution* nach [Storn 1997] statt.

Evaluation

Für die Evaluation wurde für jedes Referenzbild ein zusammengeführtes Kandidatenpaar gelabelt. Hierfür wurden mindestens vier Bildpunkte, welche möglichst nah zu den Ecken des Referenzbildes liegen, in dem zusammengeführten Kandidatenpaar gelabelt. Zwischen den vier Punkten lässt sich ein Raster aufspannen, welches zur Evaluation verwendet wird.

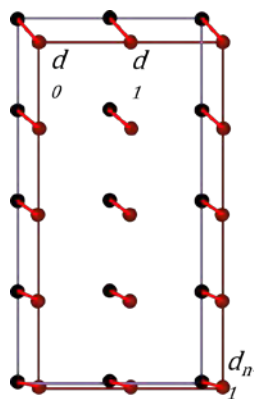


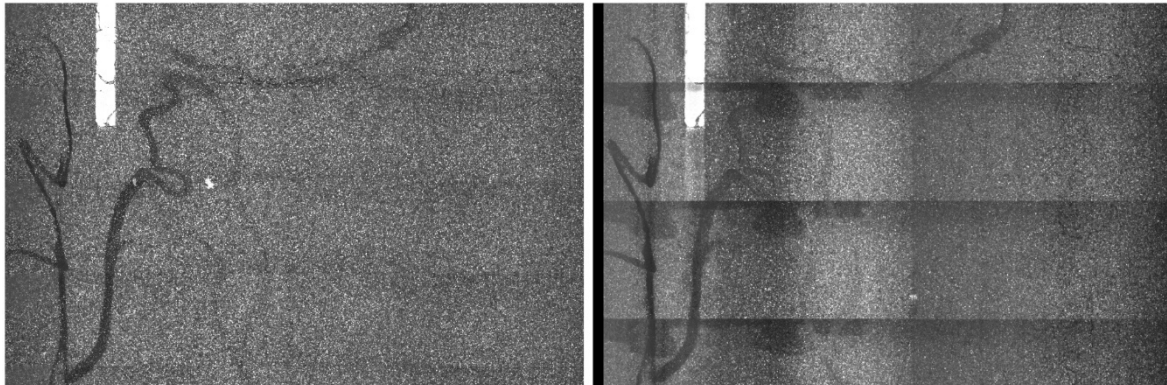
Abbildung 63: Rasterbasierte Distanzmetrik für die Bewertung der transformierten Oberfläche. [Lembach 2019]

Bei der Evaluation wird die gefundene Transformation auf das Referenzraster angewendet, wodurch ein zweites Raster mit verschobenen Punkten entsteht, Abbildung 63 zeigt dies

beispielhaft. Anschließend werden die Distanzen zwischen Referenzrasterpunkt und dem verschobenen Rasterpunkt berechnet und gemittelt.

$$D = \frac{1}{n} \sum_{i=0}^{n-1} d_j$$

Zur besseren Einschätzung dieser Distanzmetrik ist in Abbildung 64 ein Beispiel für $D=30$ dargestellt.



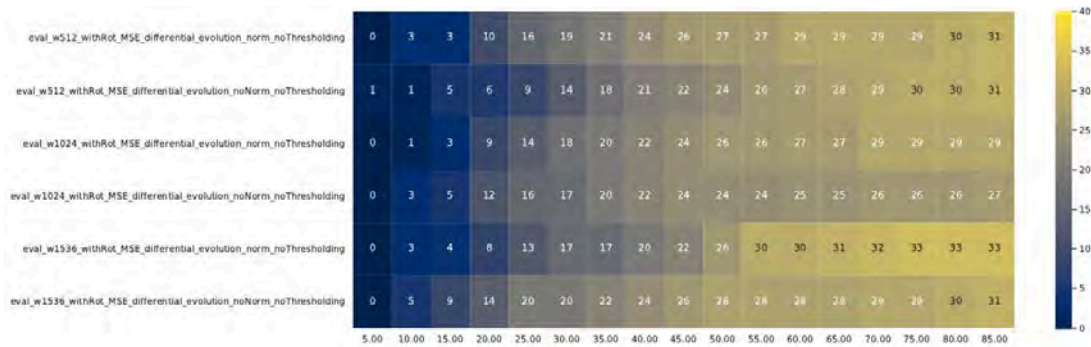
(a) Referenzbild

(b) Transformiertes Bild mit $D = 30$

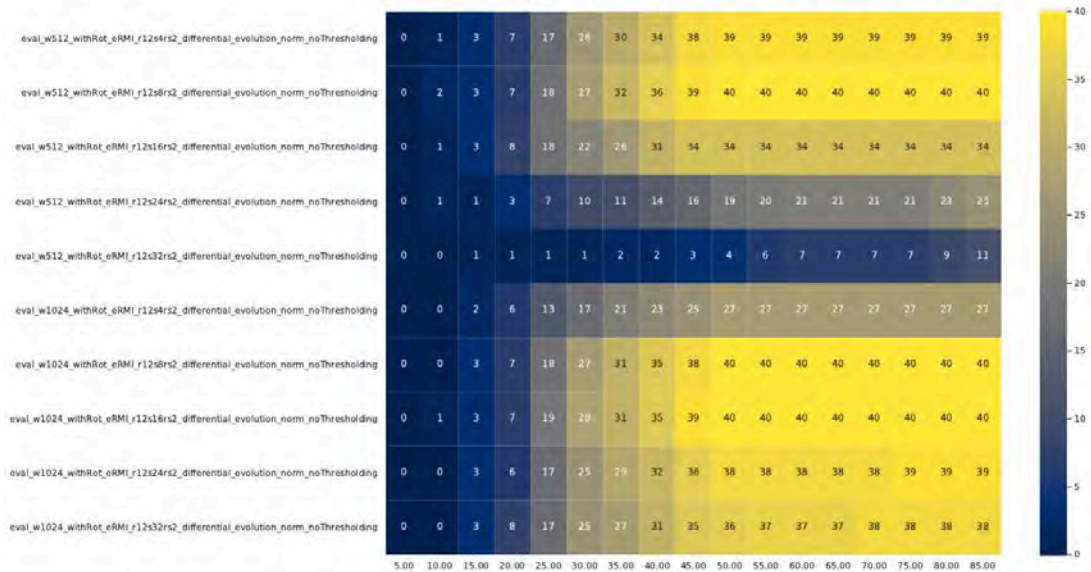
Abbildung 64: Beispiel einer zeitlichen und räumlichen Synchronisierung von Oberflächenbildern. Ein Distanzwert von $D = 30$ liefert präzise Lokalisation und Überlagerung.

Ergebnisse

Im Folgenden werden die wichtigsten Ergebnisse aus der Masterarbeit [Lembach 2019] und den darauf aufbauenden Arbeiten vorgestellt. Es hat sich gezeigt, dass in den meisten Fällen weder der Rasterausgleich, noch das Thresholding zu bedeutenden Verbesserungen geführt haben, sodass im Folgenden die Normalisierung als Vorverarbeitungsschritt genutzt und verschiedene Varianten von MSE und RMI dargestellt werden. Um die Ergebnisse konsistent zu denen aus [Lembach2019] zu halten, werden im Haupttext nur die Ergebnisse zu der Autobahn-Sequenz von 2013-2017 vorgestellt. Die Ergebnisse der weiteren Sequenzen sind im Anhang 14.1 dargestellt.



(a) MSE



(b) RMI

Abbildung 65: Autobahn (2013-17): Anzahl gefundener Transformationen mit zugehörigem Evaluationsmaß D .

Abbildung 65 zeigt die Anzahl an Transformationen, die eine mittlere Rasterdistanz D jeweils unterschreiten. Wir unterscheiden zwischen den Kostenfunktionen MSE und RMI sowie verschiedenen Parametern davon. Es zeigt sich, dass durch die Einführung von RMI als Kostenfunktion deutliche Steigerungen im Bereich $30 < D < 50$ erreicht wurden. Wie Abbildung 64 zeigt, ist eine Transformation mit $D=30$ ausreichend für eine Bewertung der Schadensentwicklung. Der Anteil an Bildern, die mit einer Genauigkeit von $D \leq 30$ synchronisiert werden können, liegt je nach Sequenz und Variante zwischen 70 und 87 Prozent. Für $D \leq 50$ liegen die Grenzwerte sogar bei 86 und 100 Prozent.

Somit können fast alle Bilder der Sequenzen sehr präzise synchronisiert werden. Dies deutet auf die Robustheit von RMI als Optimierungsmaß hin. In den weiteren Sequenzen im Anhang haben sich zusätzlich zu der hohen Robustheit auch für kleinere D Werte deutliche Steigerungen gezeigt, die auf hohe Präzision hindeuten.

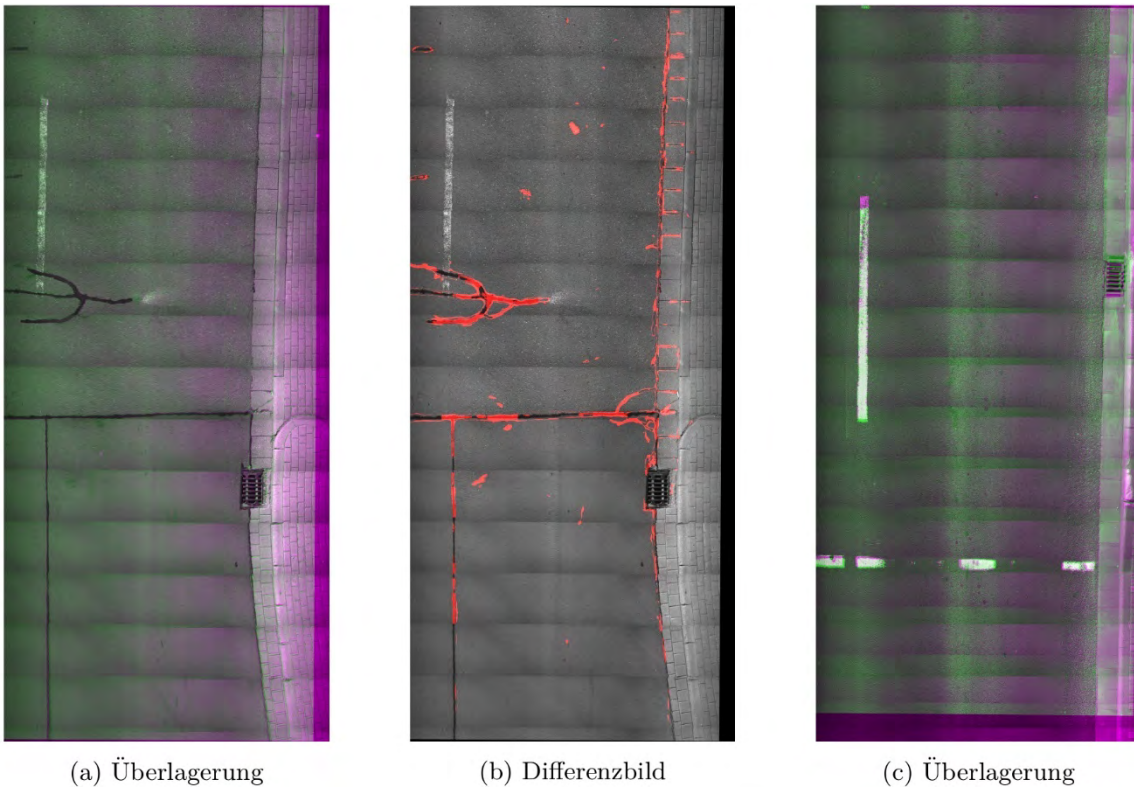


Abbildung 66: Beispiel für eine gute Synchronisation (a) und das zugehörige Differenzbild (b) sowie eine schlechtere Synchronisation (c) erkennbar an den grün- und magentafarbenen Kanten an der Fahrbahnmarkierung.

Im Verlauf des Projektes wurden noch eine Vielzahl an Varianten (vor allem von RMI) untersucht, welche allerdings den Rahmen dieses Berichts überschreiten würden. In der praktischen Anwendung wäre es sinnvoll, die jeweils optimale Variante der Kostenfunktion für das jeweilige Szenario (Straßentyp) auszuwählen.

10.1.2 Räumliche Synchronisation von Oberflächen- und Frontbildern

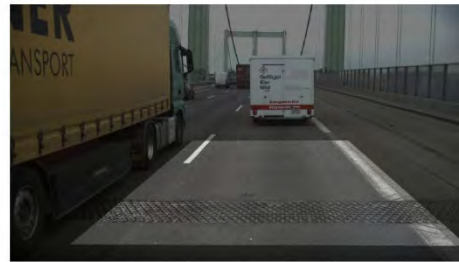
Bei dem händischen Prozess der Schadensauswertung werden, bei schweren Entscheidungen, neben dem Oberflächenbild auch Frontbilder verwendet. Es wurde daher zusätzlich untersucht, wie gut Oberflächenbilder automatisch mit Frontbildern synchronisiert werden können, damit die automatische Schadensdetektion verbessert werden kann. Eine ideale Synchronisation von Oberflächen- und Frontbild wird in Abbildung 67 präsentiert. In der Masterarbeit [Lembach2019] wurde dieser Aspekt näher untersucht. Hierfür müssen stabile, visuelle Merkmale in den Bildern gefunden werden und anhand dieser die Bilder zueinander ausgerichtet werden.



(a) Oberflächenbild



(b) Frontbild



(c) Ausgerichtet und überlagert

Abbildung 67: Beispiel für eine ideale Synchronisation von Oberflächen- zu Frontbild. In (b) wird das Frontbild zu dem Oberflächenbild aus (a) gezeigt, welche zueinander ausgerichtet werden sollen. (c) zeigt das Frontbild, welches mit dem ausgerichteten Oberflächenbild überlagert wurde. [Lembach, 2019]

Für die räumliche Synchronisation von Oberflächen- und Frontbildern, wie sie in Abbildung 67 dargestellt ist, muss eine Transformation gefunden werden, welche das Oberflächenbild in die Bildebene des Frontbildes überführt. Die gesuchte Homographie lässt sich durch vier Korrespondenzen in den beiden Bildern berechnen. In [Lembach 2019] wurden verschiedene Verfahren getestet, welche eine Homographie auf Basis vier möglichst exakter Korrespondenzen bestimmen soll. Die untersuchten Teilaspekte können in vier Gruppen aufgeteilt werden:

Vorverarbeitung: Hierbei wurde neben den Vorverarbeitungen aus Abschnitt 10.1.1 das Frontkamerabild maskiert (Teile außerhalb der Straße werden ausgeschnitten) und das Oberflächenbild in einer dem Ziel naheliegenden Pose vortransformiert. Weiterhin wurde auf das Schwellenwertverfahren verzichtet, da hierdurch die Features auf den Bildern an Aussagekraft verlieren. Weiterhin wurden die Bilder geglättet, da viele Frontkamerabilder verrauscht waren.

Detektoren: Mit diesen Verfahren werden relevante Punkte in einem Bild gesucht. Folgende Detektoren wurden getestet: Harris Corners [Harris 1988], Harris Laplace [Mikolajczyk 2005], Scale Invariant Feature Transform (SIFT) [Lowe 2004], Speeded Up Robust Features (SURF) [Bay 2006], Feature from Accelerated Segment Test (FAST) [Rosten

2006], Adaptive and Generic Accelerated Segment Test (AGAST) [Mair 2010], Binary Robust Invariant Scalable Keypoint (BRISK) [Leutenegger 2011], Oriented FAST and Rotated BRIEF (ORB) [Rublee 2011].

Deskriptoren: Mit diesen Verfahren werden Merkmalsvektoren für die Punkte, welche durch einen Detektor gefunden wurde, berechnet. Folgende Deskriptoren wurden getestet: Binary Robust Independent Elementary Features (BRIEF) [Calonder 2010], Fast Retina Keypoint (FREAK) [Alahi 2012], DAISY [Tola 2010], VGG [Simonyan2014].

Feature-Matching: Hierbei wurde üblicherweise die euklidische Distanz verwendet, um die durch die Deskriptoren berechneten Merkmalsvektoren miteinander vergleichen zu können. Für binäre Merkmalsvektor wurde die Hamming Distanz verwendet.

Bestimmen der Homographie: Als letzter Schritt muss ein Verfahren für die Berechnung der Homographie eingesetzt werden, da die bestimmten Korrespondenzen nicht perfekt sind. Hierfür wurde Random Sample Consensus (RANSAC) [Fischler1981] und Progressive Sample Consensus (PROSAC) [Chum2005] verwendet.

Ebenso wurden in [Lembach 2019] Optimierungsverfahren getestet, welche ohne das Bestimmen von vier Korrespondenzen auskommt, um eine Homographie zu bestimmen. Diese sind ähnlich zu dem in Abschnitt 10.1.1 getestetem Verfahren.

Obwohl in der Masterarbeit [Lembach 2019] ein großer Umfang an Verfahren getestet wurde, konnte keine stabile automatische Synchronisation von Oberflächen- und Frontbildern gefunden werden. Dabei wurden folgende Probleme identifiziert:

- **Auflösung:** Das Frontbild hat oft eine geringe Auflösung und hierbei nimmt der relevante Straßenabschnitt auch nur einen kleinen Teil des Bildes ein.
- **Beleuchtung:** Front- und Oberflächenbild sind oft stark unterschiedlich beleuchtet, wodurch die Synchronisation besonders erschwert wird.
- **Schärfe und Verdeckung:** Das Frontbild kann teilweise eine Unschärfe aufweisen. Zusätzlich können Bereiche im Frontbild durch andere Fahrzeuge verdeckt sein.
- **Aufnahmesynchronisation:** Insbesondere bei älteren Aufnahmen ist es nicht immer gewährleistet, dass zu einem Oberflächenbild der aufgenommene Straßenabschnitt nah im Frontbild ist oder überhaupt komplett zu sehen ist. In Anbetracht dieser schlechten Aufnahmesynchronisation wird die Synchronisation in der Nachbearbeitung zusätzlich erschwert oder in Extremfällen unmöglich. Bei neueren Befahrungen sollte dieser Fehler jedoch nicht mehr auftreten.

- **Merkmale:** Oberflächenbilder haben oft sehr wenige relevante bildliche Merkmale, welche auf dem Frontbild wiedergefunden werden können.

10.1.3 Fazit

In diesem Ergebnisbericht wurden die bisherigen Ergebnisse der räumlichen Synchronisation von Straßenbildern und die relevanten Ergebnisse der Masterarbeit [Lembach 2019] und den anschließenden Arbeiten zusammengefasst. Die in der Masterarbeit entwickelten Verfahren wurden in Bezug auf die Synchronisation von Oberflächenbildern erweitert und auf einem umfassenderen Datensatz evaluiert. Es wurde eine räumliche Synchronisation von Oberflächenbildern entwickelt und erfolgreich getestet. Der derzeitige beste Ansatz ist je nach Straßentyp in der Lage zwischen 86% und 100% der Bilder zufriedenstellend zu synchronisieren.

Ebenso wurde die Synchronisation von Oberflächen- und Frontbildern betrachtet. Die Motivation bestand darin, dass ein Frontbild bei der automatisierten Detektion von Schäden an der Straßeninfrastruktur eventuell die Detektionsleistung verbessern könnte. Es zeigte sich jedoch, dass eine solche Synchronisation nur bedingt möglich ist. Als Hauptproblem stellte sich die Aufnahmetechnik der Frontbilder heraus. Hierbei ergaben die geringe Auflösung, schlechte Beleuchtung und Schärfe, Verdeckungen der Linse und die schlechte Aufnahmesynchronisation als problematisch.

Im Rahmen des Projekts erfolge eine Publikation zu der Methode der Synchronisation in [Lembach 2020].

10.2 *Räumliche Synchronisation von Schadstellen*

Aufbauend auf der räumlichen Synchronisation und der Schadendetektion des Detektors war ursprünglich im Projekt eine Synchronisation von Schäden auf Einzelschadenebene vorgesehen. Durch Erkenntnisse und Ergebnisse aus der Bearbeitung der Arbeitspakete zur Detektion von Schäden und zur Synchronisation von Schäden, wurde eine Synchronisation einzelner Schäden, wie im Projektantrag formuliert, als nicht zielführend angesehen. Der Grund ist vor allem die Erkenntnis, dass Einflüsse, wie beispielsweise Charakteristika der Bildaufnahme (Umwelteinflüsse, Kamerasystem, Bildauflösung), die Qualität des Matchings multitemporaler Bildsequenzen und auch die Qualität der Schadensdetektion, einer eindeutigen und robusten Zuordnung von Schadens-IDs nicht zuträglich sind und die Kombination dieser Einflussfaktoren zu Fehlinterpretationen hinsichtlich der Entwicklung von Schäden führen könnten.

Als Alternative zu dem im Antrag avisierten Ansatz wurde daher im Rahmen des Projektverlaufes ein Konzept zur Quantifizierung von Schadensentwicklung erarbeitet, welches auf die Möglichkeiten, die sich aus der automatisierten Bildauswertung ergeben, aufsetzt, jedoch nicht auf der Ebene einzelner Schadensobjekte mit IDs agiert. Das im folgenden erläuterte Konzept wurde im Rahmen eines Projekttreffens durch alle Projektbeteiligten als praxisrelevant und den Zielen des Projektes angemessen akzeptiert.

10.2.1 Konzept

Grundlage für die zeitliche Analyse von Schäden bildet eine räumliche Synchronisation von zeitlich versetzt aufgenommenen Befahrungssequenzen. Aufbauend auf den räumlich synchronisierten Befahrungssequenzen und auf Grundlage der von dem Detektor generierten Schadenskarte soll die Bewertung der zeitlichen Schadensentwicklung auf Grundlage einer sogenannten *Change Detection Analyse* erfolgen. Diese Art der Analyse ist ein im Remote-Sensing-Bereich weit verbreitetes Verfahren zur Erkennung von Veränderungen in Bilddaten, die zu verschiedenen Zeitpunkten aufgenommen wurden [Fichera 2012]. Dieses Konzept wird leicht abgewandelt adaptiert, um Schadensausprägungen zu verschiedenen Zeitschnitten entlang definierter Kriterien zu vergleichen. Da noch keine explizite Richtlinie zur Quantifizierung der Entwicklung von Straßenschäden auf dem Skalenniveau einzelner Schäden existiert, wurden im Rahmen des Projektes mit den Projektbeteiligten Folgendes abgestimmt:

- Verglichen werden nicht einzelne Schäden mit IDs, sondern als Bezugseinheit dient das Eingabebild. Der Abbildungsbereich des Eingabebildes bestimmt damit den räumlichen Geltungsbereich für die Analyse von Veränderungen über die Zeit. Für die in Deutschland verwendeten Oberflächenbilder kann demnach alle 10 Meter eine Aussage über eine Schadensentwicklung getroffen werden. Für die in Österreich eingesetzten Oberflächenbilder ist dies alle 3 Meter möglich.
- Für linear ausgeprägte Schäden (z.B. offener Riss, Nahttriss oder sanierter Riss) werden als Indikatoren die Länge und Längenänderung pro Eingabebild berechnet.
- Für flächenhaft ausgeprägte Schäden (z.B. Flickstelle, Ausmagerung oder Binde-mittelaustritt) werden als Indikatoren die Fläche und Flächenänderung pro Eingabebild berechnet.

Abbildung 68 zeigt den schematischen Ablauf des Konzepts.

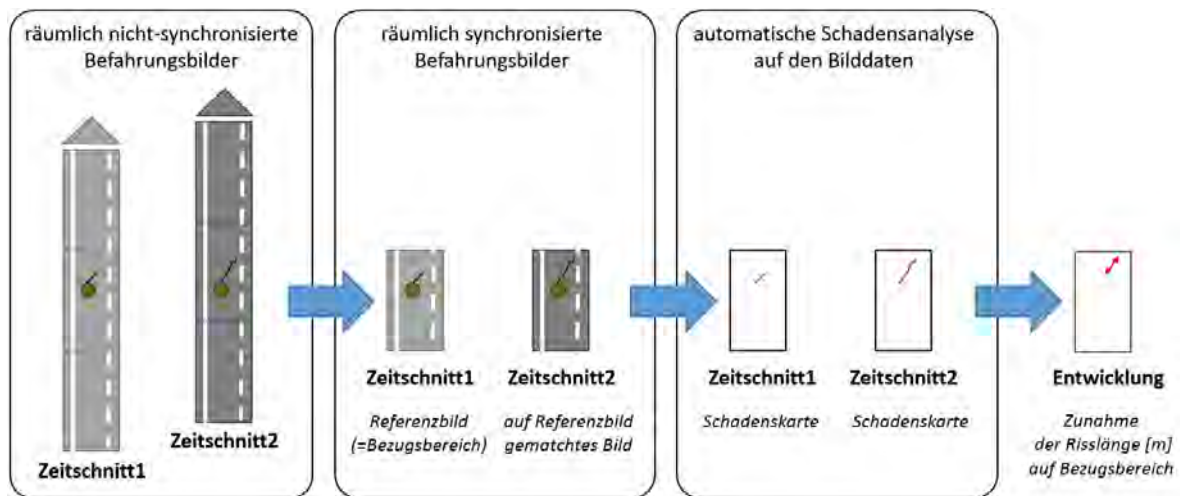


Abbildung 68: Konzept zur Bewertung der Schadenentwicklung. Die ersten drei Blöcke der Ablaufskizze sind Inhalt der Teilarbeitspakete AP6.2 und AP3 (von links nach rechts). Nach einer groben Zuordnung von Bildpaar-Kandidaten mithilfe ihrer Geo-Tags, erfolgt der Zuschnitt und das Matching auf das Referenzbild mithilfe von Bildmerkmalen (hier durch den Schachtdeckel angedeutet). Sowohl für das Referenzbild als auch das korrespondierende Bild werden vom Detektor im Zuge der automatischen Schadensanalyse Schadstellenkarten in Form von Pixelmasken erzeugt (hier angedeutet durch die rote Riss-Linie). Im letzten Schritt werden jeweils für das gesamte Bild Indikatoren für die vorherrschenden Schadensarten berechnet, die dann verglichen werden können, um die Entwicklung zwischen den beiden Zeitschnitten zu quantifizieren.

10.3 Analyse der Schadenentwicklung

Der unter 10.2 beschriebene Ansatz zur Bewertung der Schadenentwicklung wurde im Rahmen des Projekts in Form eines Software-Prototyps ausgebaut. Die entsprechenden Algorithmen und Visualisierung sind in Form eines Python Skripts (jupyter Notebook) implementiert. In Abbildung 69 ist das Interface des finalen Prototyps dargestellt. Mithilfe des Sliders am oberen Interfacerrand kann linear durch die Segmentierungsergebnisse der verglichenen Messfahrten navigiert werden. Je Bilddatenpaar werden die Indikatoren Länge[m] bzw. Fläche[m²] der relevanten Schadensmerkmale sowie in einem weiteren Diagramm die Veränderung für einzelne Schadensmerkmale angezeigt. Am unteren Interfacerrand wird das Ergebnis des multitemporalen Vergleichs in Form einer Tabelle angezeigt, welche auch als Ausgabe-Datei für externe Weiterverarbeitungszwecke exportiert werden kann.

Systemvoraussetzungen:

- Python (z.B. über Anaconda Distribution für Windows) inkl. der Entwicklungsumgebungen jupyter Notebook bzw. jupyter Lab
- diverse Python Bibliotheken (Streamlit, OpenCV, skimage, PIL, numpy, pandas, matplotlib, glob und tqdm)



- M02_B2052501_00000000_segMap.png
- M02_B2052501_00000001_segMap.png
- M02_B2052501_00000002_segMap.png
- M02_B2052501_00000003_segMap.png
- M02_B2052501_00000004_segMap.png
- M02_B2052501_00000005_segMap.png
- M02_B2052501_00000006_segMap.png
- M02_B2052501_00000007_segMap.png
- M02_B2052501_00000008_segMap.png
- M02_B2052501_00000009_segMap.png
- M02_B2052501_00000010_segMap.png



- M02_B2152703_00000000_segMap.png
- M02_B2152703_00000001_segMap.png
- M02_B2152703_00000002_segMap.png
- M02_B2152703_00000003_segMap.png
- M02_B2152703_00000004_segMap.png
- M02_B2152703_00000005_segMap.png
- M02_B2152703_00000006_segMap.png
- M02_B2152703_00000007_segMap.png
- M02_B2152703_00000008_segMap.png
- M02_B2152703_00000009_segMap.png
- M02_B2152703_00000010_segMap.png

Abbildung 70: Schematische Darstellung der erwarteten Datenablagestruktur für den Software-Prototyp. Die Ordner T1 und T2 repräsentieren zwei verschiedene Zeitschnitte. Im Ordner des jeweiligen Zeitschnitts werden die Ergebnis-Masken der automatischen Schadstellendetektion im PNG-Format erwartet. Die Zuordnung der Bilder erfolgt in diesem Schritt über die Bildnamen, diese wurden im Rahmen der räumlichen Zuordnung synchronisiert.

10.3.1 Experimentelle Analyse und Evaluierung

Für die Anwendung und Evaluierung des Konzepts zur Bewertung der Schadensentwicklung wurde im Rahmen des Projektes am 19.10.2018 und am 30.03.2019 (vor und nach Frostwechsel) ein ca. 1.3 km langer Straßenzug innerorts mit dem Messsystem S.T.I.E.R aufgenommen. Die Messungen der beiden Zeitschnitte wurden bewusst mit einem gewissen Offset in Fahrtrichtung aufgenommen. Für die unabhängigen Messungen der beiden Zeitschnitte wurden regelwerkskonforme Oberflächenbilder von 10m Länge erzeugt. Jedes 10m-Bild ist dabei durch eine Geokoordinate verortet.

Mithilfe des in 10.1 beschriebenen Ansatzes zu Bildsynchronisation wurden die Bilddatenströme der beiden Zeitschnitte in einem ersten Schritt über ihre Geoposition grob gematched und anschließend aus dem Bilddatenstrom des zweiten Zeitschnitts Oberflächenbilder von 10m Länge berechnet, die räumlich dem jeweiligen Bildinhalt des ersten Zeitschnittes entsprechen. In Abbildung 67 ist schematisch dargestellt, wie aus dem Zeitschnitt II die Bilder n und n+1 über ihre Geokoordinaten als Match-Kandidaten für die Synchronisation mit dem Bild n des Zeitschnitts I selektiert wurden. Anschließend werden aus den beiden Kandidaten die jeweiligen Bildbereiche ausgeschnitten (rote Box) und so auf das Referenzbild transformiert, dass sich die Bildinhalte möglichst gut entsprechen. Im Zuge der Transformation werden fünf Parameter geschätzt: zwei Translationen, zwei Skalierungsfaktoren und ein Rotationsfaktor.

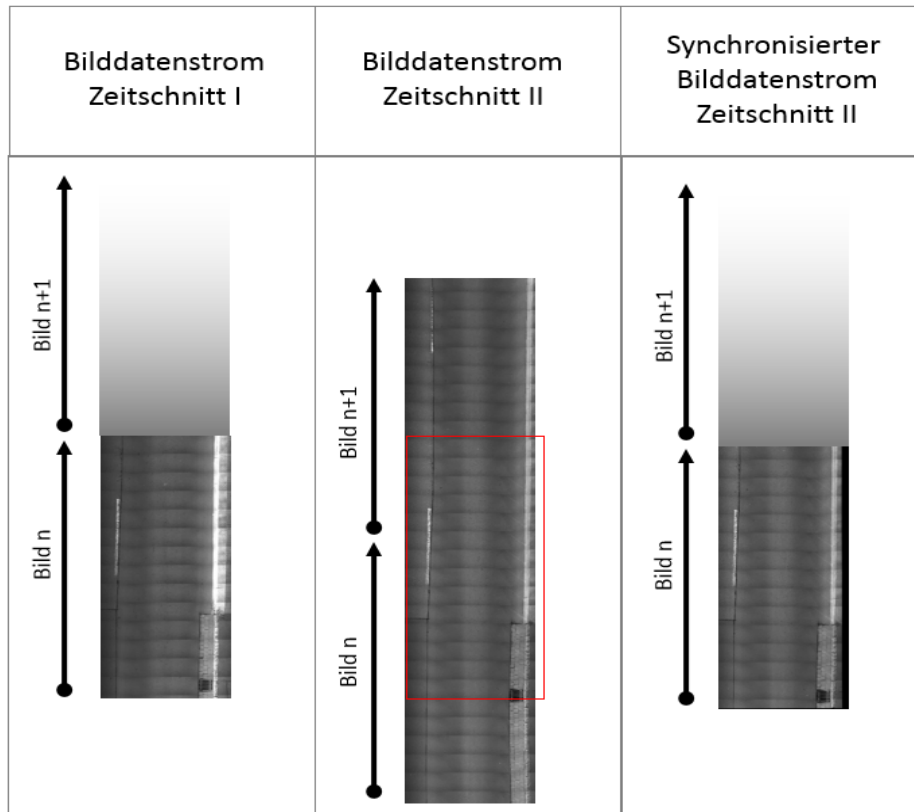
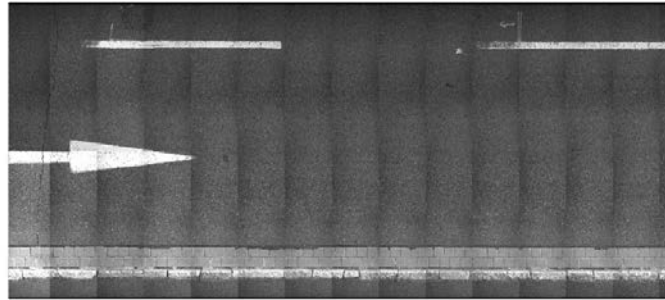


Abbildung 71: Schematische Darstellung der Bildsynchronisation zweier multitemporaler Bilddatenströme.

Nachdem die synchronisierten Oberflächenbilder der beiden Zeitschnitte vorlagen (siehe Abbildung 72), wurde die automatische Detektion angewandt und die Detektionsergebnisse von dem Modell als Pixelmasken im PNG-Format ausgegeben (siehe Abbildung 73). Dies ist eine Option bei Anwendung des Modells.

Referenz
Bilddatenstrom
Zeitschnitt: 2018



Synchronisierter
Bilddatenstrom
Zeitschnitt: 2019

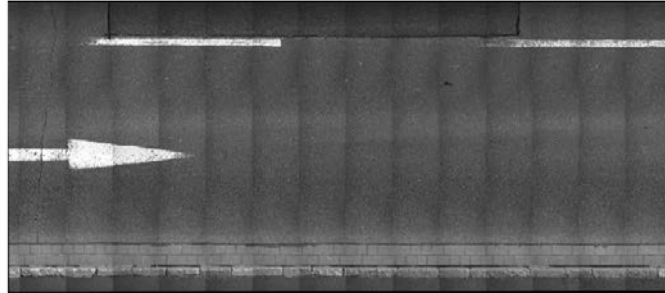


Abbildung 72: Beispiel für ein synchronisiertes Bilddatenpaar.

Segmentierungsergebnis
Referenz
Bilddatenstrom
Zeitschnitt: 2018



Segmentierungsergebnis
Synchronisierter
Bilddatenstrom
Zeitschnitt: 2019



Abbildung 73: Ergebnis der automatischen Schadstellenanalyse für das in Abbildung 72 gezeigt Bildpaar.

Mit der Software zur Analyse der Schadensentwicklung wurden abschließend diese Segmentierungsergebnisse geladen, auf deren Basis für jedes Eingabebild die Indikatoren Länge und Fläche für die relevanten Schadklassen berechnet und schließlich die Veränderungen vom Zeitschnitt 2018 zum Zeitschnitt 2019 tabellarisch in Form einer CSV-Datei ausgegeben. Das Visualisierungsskript wiederum ist in der Lage, diese CSV auszulesen und die entsprechenden Inhalte in einer GUI anzuzeigen (siehe Abbildung 69 und Abbildung 74)

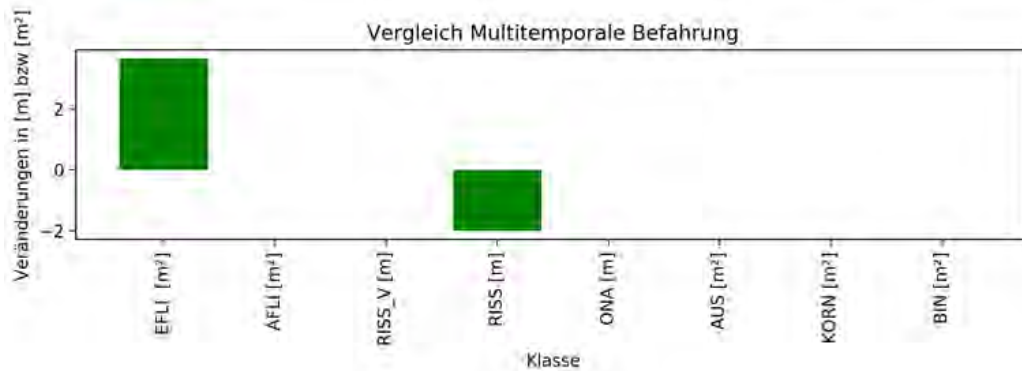


Abbildung 74: Visualisierung der Schadensentwicklung für das in Abbildung 72 gezeigte Bildpaar. Die Schadensentwicklung in diesem Beispiel ist gekennzeichnet durch eine Abnahme der Risslänge um ca. 2m sowie durch eine Zunahme der Flickstellenfläche um ca. 3.6 m².

Als Ergebnis der Evaluierung des Vergleichs der multitemporalen Schadensauswertung ist folgendes festzuhalten:

- Im untersuchten Streckenabschnitt konnte keine signifikante Veränderung einzelner Schadstellen festgestellt werden (im Sinne eines Längen-/Flächenzuwachses einer Schadstelle). Vielmehr handelt es sich bei den Veränderungen im untersuchten Streckenabschnitt um Situationen, bei denen neue Schadstellen entstanden. Dies wurde manuell verifiziert und entspricht dem tatsächlichen Schadensbild.
- Im untersuchten Streckenabschnitt konnte keine Zustandsverschlechterung festgestellt werden (im Sinne einer Ausprägung von Ausbrüchen an vormaligen Riss-Stellen).
- Im untersuchten Streckenabschnitt wurde keine Sanierung durch „Vergießen“ von Rissen durchgeführt.
- Im untersuchten Streckenabschnitt wurden bereichsweise Sanierungen durch eingelegte Flickstellen durchgeführt.
- Im untersuchten Streckenabschnitt wurden bereichsweise Deckensanierungen durchgeführt, die aufgrund der Aspekte „Länge > 20m“ und „Abdeckung der gesamten Fahrstreifenbreite“ nicht als eingelegte Flickstelle (EFLI) zu bewerten sind. Da die automatische Schadstellendetektion nur die Information eines Bildes (ohne Vorgänger/Nachfolger) berücksichtigt, ist in der Regel Beginn und Ende der Deckensanierung als EFLI fehldetektiert.

Darüber hinaus ist festzuhalten, dass diverse Einflussfaktoren zu einer Verzerrung der Aussagekraft der berechneten Flächen- und Längenänderungen führen können. Nachfolgend ist beschrieben, welche Aspekte wie wirken können.

- **Einflussfaktor Sichtfeld/Verdeckungen:** Da die Aufnahme von Oberflächenbildern im fließenden Verkehr stattfindet, kann es vorkommen, dass Randbereiche des Bildes durch z.B. Verkehrsteilnehmer oder abgestellte Fahrzeuge verdeckt sind. Auch kleine Abweichungen von der Fahrlinie zwischen zwei Zeitschnitten können zur Folge haben, dass leicht unterschiedliche Bereiche der Fahrbahnoberfläche durch das Kamerasystem abgebildet sind. Wird beispielsweise ein Riss, der in Zeitschnitt 1 vorhanden ist, durch ein Fahrzeug in Zeitschnitt 2 verdeckt, würde das zu einer Abnahme der Risslänge führen und zu einer falschen Interpretation der Schadenentwicklung.
- **Einflussfaktor Bildqualität, Aufnahmesystem und Qualität der automatischen Analyse:** Da zwischen zwei Zeitschnitten oft mehrere Jahre liegen und dieselbe Strecke nicht zwingend vom selben Dienstleister mit derselben Sensorik aufgenommen werden muss, können die Oberflächenbilder unter Umständen hinsichtlich der Aspekte Auflösung, Ausleuchtung, Schatten oder gar dem Aufnahmeverfahren (z.B. Zeilenkamera vs. Flächenkamera) differieren. Auch wenn die trainierten Detektoren zu einem gewissen Grad in der Lage sind, zu generalisieren, sind Qualitätsunterschiede in der Detektionsleistung in Abhängigkeit vom verwendeten Eingabe-Bildmaterial vorhanden. Wird beispielsweise ein Riss in Zeitschnitt 1 nur unvollständig, derselbe tatsächlich unveränderte Riss in Zeitschnitt 2 jedoch vollständig detektiert, dann würde das zu einer Zunahme der Risslänge führen und zu einer falschen Interpretation der Schadenentwicklung, nur aufgrund der besseren Leistungsfähigkeit des Detektors auf dem Bildmaterial von Zeitschnitt 2.
- **Einflussfaktor Qualität des Bild-Matchings:** Kern der Bildsynchronisation (Matching) ist eine Transformation. Diese liefert immer dann gute Ergebnisse, wenn es robuste Merkmale auf dem zu matchenden Bildmaterial gibt, anhand derer sich entsprechende Bildbereiche gut aufeinander referenzieren lassen. In Bereichen, in denen diese Merkmale fehlen, wird auch die Bildsynchronisation nur eingeschränkt gute Ergebnisse liefern. In der Folge können beispielsweise Bildbereiche verzerrt/skaliert transformiert oder in seltenen Fällen partiell falsch zugeordnet sein. Ein Riss, der sich zwischen den Zeitschnitten 1 und 2 in seiner Länge tatsächlich nicht verändert hat, kann so einzig aufgrund einer fehlerhaften Transformation in Zeitschnitt 2 länger geworden sein. Dies würde zu einer Zunahme der Risslänge führen und zu einer falschen Interpretation der Schadensentwicklung.

10.3.2 Fazit

Für die Arbeiten zum Thema „Bewertung der Schadensentwicklung“ wurde ein Ansatz gewählt, der an *Change Detection* Verfahren aus dem Fernerkundungsbereich angelehnt ist. Grundvoraussetzung ist daher, dass die multitemporalen Datensätze räumlich synchronisiert vorliegen. Im Unterschied zu konventionellen *Change Detection* Verfahren wurde auf einen pixelweisen Vergleich der multitemporalen Datensätze verzichtet. Das entwickelte Analysekonzept vergleicht stattdessen Indikatoren, die ein multitemporales Bilddatenpaar im Hinblick auf etwaig auftretende Schäden entlang der Indikatoren Länge bzw. Fläche beschreibt. Die Bewertung der Schadensentwicklung geschieht somit immer in Bezug auf ein synchronisiertes Oberflächenbild-Paar. Für Deutschland sind aufgrund des Abbildungsbereichs der Oberflächenbilder alle 10m Aussagen über Zu- oder Abnahme von Schadensflächen oder – längen möglich. Für Österreich entsprechend für alle 3m.

Mit dem modifizierten *Change-Detection*-Ansatz konnte aufgezeigt werden, wie die Ergebnisse einer pixelgenauen automatischen Schadenserkenkung nach vorheriger Bildsynchronisierung zur Beschreibung der Schadensentwicklung genutzt werden können. Anhand der gewählten Indikatoren kann die Entwicklung von Schäden qualifiziert und quantifiziert werden, wobei diverse Einflussfaktoren zu einer Verzerrung der Aussagekraft der berechneten Flächen- und Längenänderungen führen können. Trotz der vielversprechenden ersten Ergebnisse bietet das Themenfeld „Schadensentwicklung“ Raum für weitere Forschungs- und Entwicklungsarbeit.

11 Übertragbarkeit auf andere Anwendungsdomänen

Weiterhin wurden die bisher im Projekt ASFaLT untersuchten Konzepte auf einer neuen Anwendungsdomäne der Straßeninfrastruktur getestet und dabei eine Machbarkeitsstudie zur Übertragbarkeit von Teilkomponenten und des Gesamtsystems auf diese neue Anwendungsdomäne durchgeführt.

11.1 Anwendungsdomäne und Datenlage

Zu Beginn des Projekts ASFaLT wurde die Domäne der Tunnelscans für die Untersuchung auf Übertragbarkeit gewählt, da nur aus dieser Domäne genügend Daten im Rahmen dieses Projekts zur Verfügung standen. Im Rahmen von ASFaLT konnten Tunnelscans von zwei Quellen für die Untersuchung auf Übertragbarkeit bezogen werden. So wurden Tunnelscans des Unternehmens Amberg Technologies und der ASFINAG (Autobahnen- und

Schnellstraßen-Finanzierungs-Aktiengesellschaft) zur Verfügung gestellt. Die Qualität und die Zweckdienlichkeit der Daten für die Untersuchung auf Übertragbarkeit werden im Folgenden näher beschrieben.

Tunnelscans der ASFINAG

Die ASFINAG hat von der reinen Datenmenge viele Daten für das Projekt bereitgestellt. Jedoch war eine Verwendung der Daten für die Untersuchung auf Übertragbarkeit nicht möglich, da sich für eine zielgerichtete Verwendung viele Probleme ergeben haben, welche im zeitlichen Rahmen der Untersuchung auf eine Übertragbarkeit nicht gelöst werden konnten.

So waren zum einen die Bilder nur in einer für das Zielsystem unüblichen Auflösung (z.B. 6800 x 250) verfügbar. Für die Weiterverarbeitung hätten die zeitlich aufeinander folgenden Bilder zusammengesucht werden müssen und daraufhin so zusammengesetzt werden müssen, sodass möglichst keine Stitchingkanten entstehen. Weiterhin waren die zur Verfügung gestellten Label schwer interpretierbar. So schien der leicht zugängliche Teil der Trainingslabel ausschließlich aus VOID-Markierungen und nicht VOID-Markierungen zu bestehen. Die für das Training benötigten Label von verschiedenen Zustandsklassen (z.B. Riss, Normal) schienen in einem proprietären Format der ASFINAG hinterlegt worden zu sein.

Eine Lösung mit vertretbarem zeitlichem Aufwand für die Extraktion der Labelinformationen aus dem proprietären Format wurde im Rahmen von ASFaLT nicht gefunden. Aus diesen Gründen wurde für die Untersuchung auf Übertragbarkeit davon abgesehen, die Tunnelscans der ASFINAG zu verwenden.

Tunnelscans aus weiteren Quellen

Zu Beginn des Projekts ASFaLT hat das Unternehmen Dibat Messtechnik GmbH eine kleine Menge an Tunneldaten dem Projekt ASFaLT zur Verfügung gestellt. Zu diesen Daten gab es zusätzlich positive Anzeichen, dass eine offizielle Kooperation, in welcher weitere Daten bereitgestellt werden könnten, mit dem Projekt ASFaLT zustande kommen könnte. Infolgedessen wurden innerhalb des Projekts ASFaLT die zunächst bereitgestellten Daten ausgewertet, verarbeitet und auf die Qualität und Zweckdienlichkeit überprüft. Jedoch kam in Folge von internen strategischen Erwägungen seitens Dibat keine Kooperation mit dem Projekt ASFaLT zustande und somit konnten die bereits vorbereiteten Daten nicht verwendet werden.

Tunnelscans des Unternehmens Amberg Technologies

Im Rahmen Projekts ASINVOS (Assistierendes und Interaktiv lernfähiges Videoinspektionsystem für Oberflächenstrukturen am Beispiel von Straßenbelägen und Rohrleitungen), welches das Vorgängerprojekt zu ASFaLT war, wurden bereits Tunnelscans durch das Unternehmen Amberg Technologies bereitgestellt. Diese Tunnelscans konnten nun im Rahmen von ASFaLT weiterhin genutzt werden. Die ursprünglichen Daten setzen sich aus vier Aufnahmen von jeweils drei Tunnel zusammen und umfassen eine Menge von 359 Bildern. Zu diesen Bildern gab es zusätzlich Labels, auf welchen Riss- und VOID-Klassen sehr grob gelabelt wurden. Diese Labels wurden ursprünglich im Projekt ASINVOS für einen Patch-basierten Ansatz verwendet.

Wie sich bei der Untersuchung auf Übertragbarkeit herausstellte, waren diese Art von Labels völlig ungeeignet für den segmentierungsbasierten Ansatz im Projekt ASFaLT. Aus diesem Grund wurde ein kleiner Teil der Bilder von der LEHMANN + PARTNER GmbH nachgelabelt. Hierbei wurde dabei darauf geachtet, dass bei dieser Labelarbeit ähnliche Konventionen eingehalten werden, wie auch bei der Labelarbeit auf den Straßendaten. So wurden nicht nur Riss- und VOID-Label vergeben, sondern auch für das System weitere relevante Klassen gelabelt. Beispielsweise befanden sich unter den weiteren gelabelten Klassen intakte Betonoberflächen, sanierte Risse, Einbauten und flächenhafte Sanierungen. Ebenfalls wurden die Bilder beinahe pixelgenau, wie auch schon bei den Straßendaten, gelabelt. So wurden engliegende Polygone um die verschiedenen Risse einzeln gezogen. Im Vergleich dazu wurde für die ursprünglich verfügbaren Labels grob ein Rechteck um eine Menge von Rissen gezogen. Durch ein solches Vorgehen befindet sich folglicherweise auch eine große Menge an Pixeln, welche der intakten Betonoberfläche zugeordnet werden müssten. Diese würden jedoch bei einem Segmentierungsansatz entweder nicht als intakte Betonoberfläche gelernt werden oder sie würden bei einem Test falsch bewertet werden.

Insgesamt wurden 16 Bilder, welche etwa die Größen 2000 x 5120 bzw. 1980 x 2930 Pixel haben, ähnlich zu den in Projekt ASFaLT festgelegten Konventionen gelabelt. Diese Bilder wurden für die Machbarkeitsstudie zu 14 Bildern für das Training und 2 Bildern für die Validierung aufgeteilt. Dabei entstammen die Trainingsbilder aus den drei verschiedenen Tun-

neln, welche in einer manuellen Auswertung völlig verschieden wirken. Weiterhin entstammen die beiden Validierungsbilder aus jeweils zwei dieser Tunnel. Im Folgenden werden die Experimente und die Ergebnisse der Machbarkeitsstudie zusammengefasst.

11.2 Experimente und Ergebnisse der Machbarkeitsstudie

Wie im vorhergehenden Abschnitt beschrieben wurden für die Machbarkeitsstudie die nachgelabelten Daten des Unternehmens Amberg Technologies verwendet. In den Experimenten wurden diese Trainingsdaten verwendet und die Experimente ähnlich zu den Experimenten aus Abschnitt 4.4 wiederholt. Hierbei wurden die Netzwerke, Architekturen, Trainingssetups und Trainingstechniken verwendet und getestet, welche sich beim Training auf Straßendaten als am vielversprechendsten herausgestellt haben. Dabei wurden jeweils mit verschiedenen Einstellungen in etwa 92 Modellen erfolgreich trainiert und evaluiert.

Experimente

Es zeigte sich, dass auch bei den Tunneldaten die besonders mächtigen, aber auch rechenaufwändigen Netzwerkarchitekturen die besten Ergebnisse erzielten. So erzielte das DeepLabV3+ mit einem Xception-Net als Backbone auch bei den Tunneldaten die besten Ergebnisse. Weiterhin zeigte sich auch, dass der Einsatz von Focal Loss vor allem bei so wenigen Trainingsbeispielen essenziell war. Ohne den Einsatz von Focal Loss wurden besonders die fein gelabelten Risse oft nicht detektiert oder aber das Modell neigte zu vielen falsch positiven Detektionen.

Evaluation

Wie zuvor beschrieben, konnte die Evaluation aufgrund der Datenlage nur auf sehr wenigen Bildern durchgeführt werden, so gibt es nur zwei Validierungsbilder und keine Testbilder. Aus diesem Grund ist die folgende Bewertung auch nur als Abschätzung der Übertragbarkeit des Gesamtsystems zu verstehen.

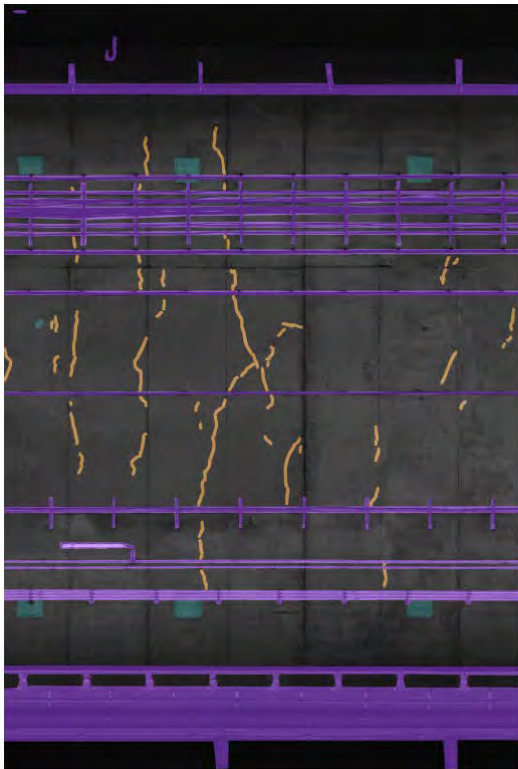
Es scheint, dass das Gesamtsystem für Straßendaten in der Lage ist, auf Tunneldaten trainiert werden zu können und dabei Informationen gelernt werden können. Ebenfalls haben sich positive Tendenzen gezeigt, dass, wenn die Tunneldaten ähnlich zu den Straßendaten aufgearbeitet werden und für das Training verwendet werden, ähnlich gute Ergebnisse erzielt werden könnten. Bei der manuellen Auswertung der derzeitigen Ergebnisse sind offensichtlich noch viele Diskrepanzen vorhanden. So gibt es trotz verschiedenster Bemühungen noch einige Fehldetektionen von Rissen auf Oberflächen, für welche das System

beim Training nicht genügend Beispiele bekommen hat. Vergleichbare Probleme sind auch beim Training von Straßendaten zu Beginn des Projekts ASFaLT aufgetreten. Diese ließen sich beheben, indem mehr Trainingsbeispiele gelabelt wurden und zum Training hinzugefügt wurden, in welchen diese Oberflächenarten vorhanden waren.

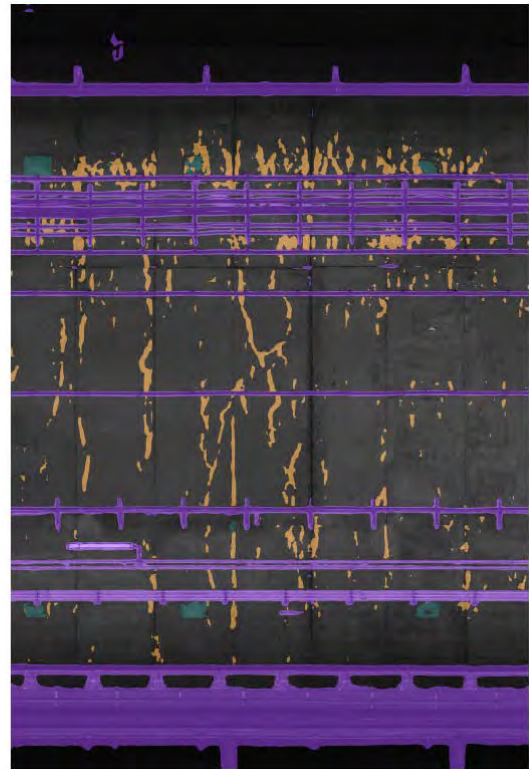
Weiterhin wurde versucht, das System nur anhand der ursprünglichen Patch-basierten Label des Unternehmens Amberg Technologies zu evaluieren und somit auf weiteren Testbildern zu testen. Hierbei ergaben sich jedoch einige Probleme. Offensichtlich sind in den ursprünglichen Labels die weiteren Klassen, welche in den neuen Labels vorhanden sind und von dem System gelernt wurden, nicht vorhanden und daher können diese auch nicht ausgewertet werden. Weiterhin ergab sich, dass die Label nicht genau auf die bereitgestellten Bilder passten und die Label versetzt waren, wobei der Offset nicht regelmäßig gleich auf allen Bildern war und so sich nicht automatisch erfassen ließ. Ebenfalls haben sich die Patches für Normalklasse und Rissklasse geschnitten. So ergaben sich zwei verschiedene Label für verschiedene Pixel, welche ebenfalls für die Auswertung nicht verwendet werden konnten. Zuletzt scheinen aber auch die vergebenen Patch-basierten Labels von mäßiger Qualität zu sein. So scheinen oft feine vereinzelte Risse, welche vom Netz detektiert werden, nicht gelabelt worden zu sein oder Bilder teilweise keine Label zu besitzen, obwohl offensichtliche Mängel auf der Aufnahme vorhanden sind. Aus diesen Gründen konnte eine weitere Auswertung des Gesamtsystems mit den ursprünglichen Patch-basierten Labels nicht abschließend durchgeführt werden.

Visualisierung der Ergebnisse

Abschließend sind in den folgenden Abbildungen die Label und Ausgaben der zwei Validierungsbilder für das Netzwerk mit den besten Validierungsergebnissen nach mIoU aufgeführt.

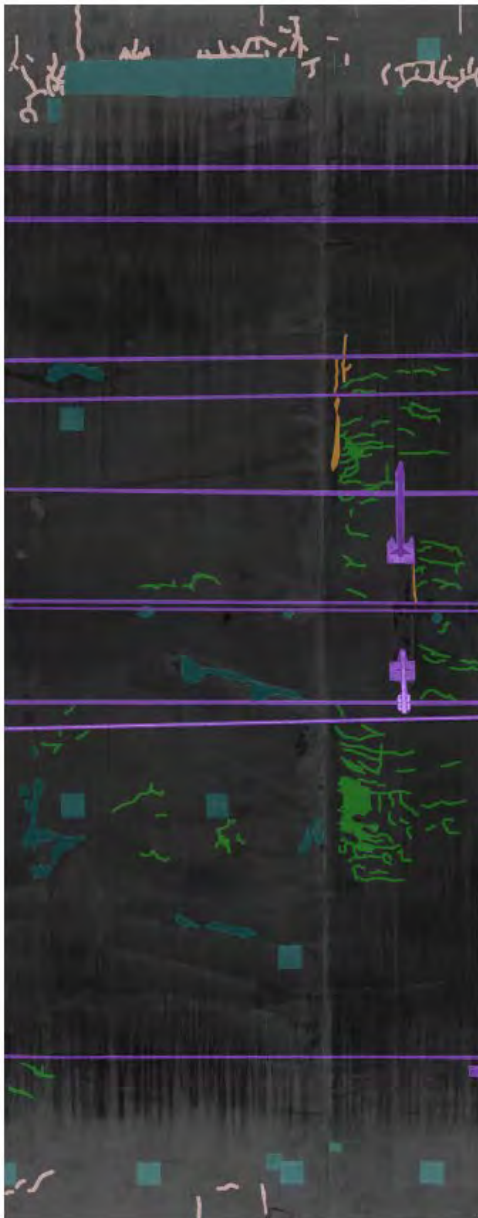


(a) Label

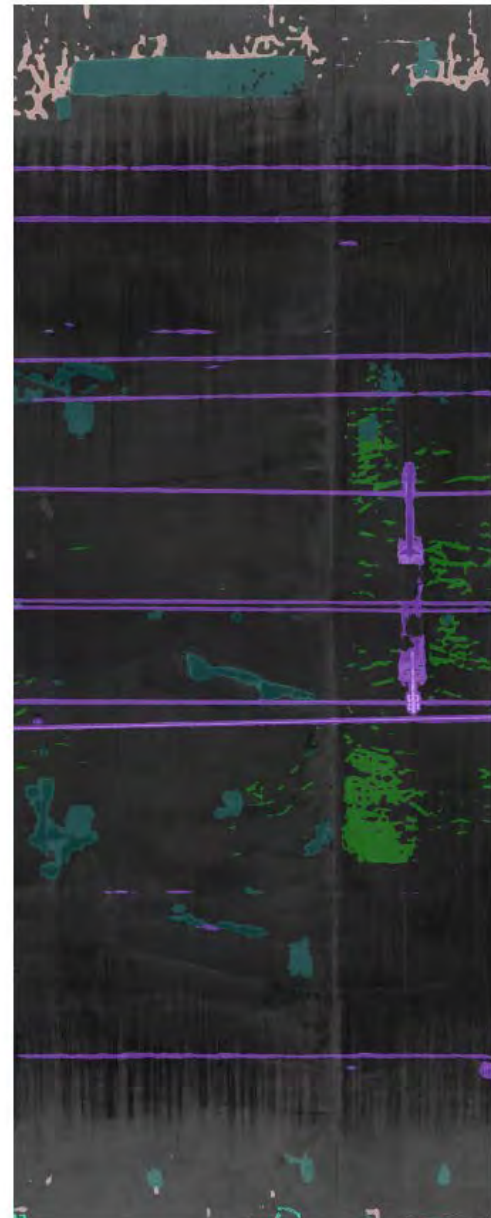


(b) Ausgabe

Abbildung 75 Diese Abbildung zeigt das Label und die Netzwerkausgabe für das erste Validierungsbild. Farbenlegende: Schwarz-Normal, Lila-Einbauten, Dunkel Grün-Flächenhafte Sanierung, Orange-Offener Riss



(a) Label



(b) Ausgabe

Abbildung 76 Diese Abbildung zeigt das Label und die Netzwerkausgabe für das zweite Validierungsbild. Farbenlegende: Schwarz-Normal, Lila-Einbauten, Dunkel Grün-Flächenhafte Sanierung, Hell Grün-Offener Riss (undeutlich), Orange-Offener Riss, Beige-Sanierter Riss

12 Fazit

Im Rahmen des Projekts ASFaLT konnte eine sehr breite Spanne an Problemfeldern für die Automatisierung und Verbesserung der Erfassung von Schäden auf Oberflächenbildern der Straße untersucht werden.

Begonnen wurde mit der systematischen Analyse und Vereinheitlichung der in den D-A-CH Ländern genutzten Standards und Vorschriften für die Bewertung von Straßenoberflächen. Während der Diskussion zwischen den Ländern und unter Einbezug der Praxis konnten viele Erkenntnisse über Gemeinsamkeiten und Unterschiede und ihre Bedeutung für die Automatisierung gewonnen werden. Der in dem Projekt entwickelte gemeinsame Labelkatalog und die länderspezifischen Schadenkataloge bieten dabei ein gutes Werkzeug, um die Automatisierung mit definierten Standardklassen durchzuführen, die sich anschließend wieder in die Regelwerke der einzelnen Länder überführen lassen.

Aufbauend auf diesem Labelkatalog wurden im Projekt eine große Anzahl an Beton- und Asphaltbildern mit pixelgenauen Labelmasken versehen, so dass das Training von Segmentierungsnetzwerken für die Schadstellendetektion ermöglicht wird.

Im Rahmen des Trainings von Detektoren wurde im Projekt eine Vielzahl von unterschiedlichen Netzwerkarchitekturen, Trainingstechniken und Parametrierungen analysiert, um für die Domäne der Schadensdetektion auf der Straße geeignete Methoden zu entwickeln und zu optimieren. Für den entwickelten Detektor auf Asphaltoberflächen konnte exemplarisch gezeigt werden, dass sich die automatisierte Segmentierung mit den Entscheidungen von Experten auf dem Gebiet der Schadenserfassung messen lassen können.

Weiterhin konnten Methoden für die Beschreibung von Oberflächeneigenschaften entwickelt und getestet werden, die sowohl eine Unterscheidung zwischen Asphalt- und Betonoberflächen ermöglichen, aber in Nachfolgeprojekten auch die Unterscheidung zwischen weiteren Subgruppen der Belagarten ermöglichen können.

Im Rahmen der Automatisierung spielt die Schätzung von Sicherheiten eine große Rolle, um die Kontrolle über die Automatisierung gewährleisten zu können. Im Rahmen des Projekts konnten daher intensive Untersuchungen in Bezug auf die Schätzung von Unsicherheiten auf synthetischen Daten und auf Straßenoberflächen durchgeführt werden. Auch hierfür wurden Methoden entwickelt, die die Schätzung von großflächigen Unsicherheiten ermöglichen und auch für die Auswahl von Trainingsdaten im Projekt angewendet werden konnten.

Für die Nutzbarkeit und Anwendbarkeit der entwickelten Lösungen wurden abschließend Module für die Konvertierung der Detektionen in die länderspezifischen Standards

umgesetzt und alle entwickelten Methoden in einem Demonstrator integriert. Im Rahmen von Nutzertests konnte dieser Demonstrator getestet werden und setzt die vollautomatische Detektion und länderspezifische Konvertierung für Oberflächenbilder für den ZEB- und den RVS Standard um.

Die von der automatisierten Detektion generierten Segmentierungskarten liefern eine Bewertung auf einem deutlich detaillierteren Niveau als die bisherigen händischen Erfassungen und liefern präzisere Angaben in Bezug auf Positionen, Flächen und Ausdehnungen von Schäden. Im Rahmen des Projekts wurde daher auch analysiert, wie eine zukünftige Bewertung der Schadensentwicklung diesen neuen Detailgrad möglichst gut nutzbar machen kann. Beginnend mit der automatisierten Überlagerung von Oberflächenbildern aus verschiedenen Befahrungen, wurde ein Konzept entwickelt, welches die Entwicklung von Schäden auf einzelnen 10-Meter-Bildern ermöglicht.

Abschließend wurde im Projekt untersucht, wie sich das entwickelte System auch auf andere Anwendungsdomänen übertragen lässt. Am Beispiel der Domäne Tunnel wurde dabei mit wenigen Trainingsbeispielen eine Demonstration der Übertragbarkeit durchgeführt.

Durch das Projekt ASFaLT konnten viele Aspekte der Automatisierung der Schadstellendetektion erfolgreich analysiert und bearbeitet werden. Für die Weiterentwicklung und Verbesserung der entwickelten Methoden scheint nach aktuellem Stand die Integration von weiteren Trainingsdaten den größten Gewinn bringen zu können. Außerdem sollte zukünftig untersucht werden, welche zusätzlichen Informationen aus der automatisierten Detektion gewonnen werden können, um die Zustandsbewertung weiter zu verbessern. Es wäre beispielsweise denkbar, auch die von dem System erkannten Fahrbahnmarkierungen genauer zu analysieren, um beispielsweise Aussagen zur deren Abnutzung treffen zu können. Auch könnte die Automatisierung für die genaue Lokalisation von Straßeninventar genutzt werden.

Offene Forschungsfragen

Mit den im Rahmen des Projekts ASFaLT gelabelten Bildern konnten für Asphalt- und Betonoberflächen geeignete Detektoren erzeugt werden, welche insbesondere auf Asphalt bereits gute und praxisrelevante Ergebnisse erzielen können. Für den systematischen Einsatz im Rahmen der standardisierten Zustandserfassung wird jedoch das Labeling weiterer Daten, mit bisher nicht repräsentierten Situationen, unumgänglich sein. Weiterhin konnte

die robuste Detektion von Flickstellen noch nicht vollumfänglich umgesetzt werden. Zur weiteren Verbesserung könnte beispielsweise der umgesetzte Ansatz zur Bestimmung der optischen Oberflächeneigenschaften genutzt werden, um eine bessere Unterscheidung zwischen Flickstellen und der normalen Fahrbahnoberfläche durchzuführen.

In diesem Zusammenhang ist auch ein umfangreicher und systematischer Vergleich der automatisierten Detektion mit der manuellen Auswertung erforderlich. Hierbei sollte auch die Qualität und Reproduzierbarkeit der manuellen Auswertung betrachtet werden, um die Vorteile und Möglichkeiten der Automatisierung einschätzen zu können.

Für die Beurteilung der Schadensentwicklung konnten bereits gute erste Ergebnisse erzielt werden. Aufbauend auf diesen Arbeiten ist weitere Forschung nötig, um die Tragfähigkeit des Konzepts in der Praxis zu evaluieren und um die softwaretechnische Integration für den Einsatz in der Praxis zu verbessern.

13 Literaturverzeichnis

- [Alahi 2012] Alahi, Alexandre, R. Ortiz und P. Vandergheynst, „FREAK: Fast Retina Key-point“. IEEE Conference on Computer Vision and Pattern Recognition, 2012.
- [Amhaz 2016] R. Amhaz, S. Chambon, J. Idier, and V. Baltazart, “Automatic crack detection on two-dimensional pavement images: An algorithm based on minimal path selection,” IEEE Transactions on Intelligent Transportation Systems, vol. 17, no. 10, pp. 2718–2729, 2016.
- [Bay 2006] Bay, Herbert, T. Tuytelaars und L. V. Gool, “SURF : Speeded Up Robust Features”. European Conference on Computer Vision ECCV 2006, S. 404–417, 2006.
- [Calonder 2010] Calonder, Michael, V. Lepetit, C. Strecha und P. Fua, “BRIEF: Binary Robust Independent Elementary Features”. European Conference on Computer Vision - ECCV 2010, 6314:778–798, 2010.
- [Chambon 2011] S. Chambon and J. M. Moliard, “Automatic road pavement assessment with image processing: Review and comparison,” International Journal of Geophysics, vol. 2011, 2011.
- [Chen 1988] Chen, Yung-Sheng und W.-H. Hsu, “A modified fast parallel algorithm for thinning digital patterns”. Pattern Recognition Letters, 7(2):99 – 106, 1988.
- [Chen 2017] Chen, Liang-Chieh, G. Papandreou, F. Schroff und H. Adam, „Rethinking Atrous Convolution for Semantic Image Segmentation“. Computing Research Repository (CoRR), abs/1706.05587, 2017.
- [Chollet 2017] Chollet, Francois, “Xception: Deep Learning With Depthwise Separable Convolutions.” In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.
- [Chum 2005] Chum, Ondrej und J. Matas, “Matching with PROSAC - Progressive Sample Consensus”. CVPR, 2005.
- [Eisenbach 2015] M. Eisenbach, A. Kolarow, A. Vorndran, J. Niebling, H.-M. Gross, “Evaluation of Multi Feature Fusion at Score-Level for Appearance-based Person Re-Identification,” Int. Joint Conf. on Neural Networks (IJCNN), pp. 469-476, 2015.
- [Eisenbach 2017] M. Eisenbach, R. Stricker, D. Seichter, K. Amende, K. Debes, M. Sesselmann, D. Ebersbach, U. Stöckert, H.-M. Gross, „How to Get Pavement Distress Detection Ready for Deep Learning? A Systematic Approach”, Int. Joint Conf. on Neural Networks (IJCNN), Anchorage, USA, pp. 2039-2047, IEEE 2017.
- [Engelhardt 2019] Engelhardt, Marius, „Deep-learning-basierte semantische Segmentierung von Strassen zur pixelgenauen Detektion von Schadstellen“. Bachelorarbeit, TU Ilmenau, 2019.
- [Fan 2018] Z. Fan, Y. Wu, J. Lu, and W. Li, “Automatic Pavement Crack Detection Based on Structured Prediction with the Convolutional Neural Network,” ArXiv e-prints, Feb. 2018.

- [Fichera 2012]** Fichera, Carmelo Riccardo, G. Modica und M. Pollino, “Land Cover classification and change-detection analysis using multi-temporal remote sensed imagery and landscape metrics”. *European Journal of Remote Sensing*, 45(1):1–18, 2012.
- [Fischler 1981]** Fischler, Martin A und R. C. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography”. *Communications of the ACM*, 24(6):381–395, 1981.
- [Gal 2016]** Y. Gal, “Uncertainty in deep learning,” Ph.D. dissertation, University of Cambridge, 2016.
- [Gal 2016b]** Gal, Yarin und Z. Ghahramani, „Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In: *Proceedings of the 33rd International Conference on Machine Learning (ICML)*.
- [Gelfand und Smith, 1990]** Gelfand, Alan E und A. F. Smith, „Sampling-based approaches to calculating marginal densities”. *Journal of the American statistical association (JASASS)*, 85(410):398–409.
- [Goodfellow 2015]** I.J. Goodfellow, J. Shlens, C. Szegedy, “Explaining and Harnessing Adversarial Examples,” *Int. Conf. on Learning Representations (ICLR)*, 2015.
- [Gopalakrishnan 2017]** K. Gopalakrishnan, S.K. Khaitan, A. Choudhary, and A. Agrawal, “Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection,” *Construction and Building Materials*, vol. 157, pp. 322–330, 2017.
- [Guo 2017]** C. Guo, G. Pleiss, Y. Sun, K.Q. Weinberger, “On Calibration of Modern Neural Networks,” *Int. Conf. on Machine Learning (ICML)*, 2017.
- [Harris 1988]** Harris, Chris und M. Stephens, “A Combined Corner and Edge Detector”. *Proceedings of Fourth Alvey Vision Conference*, S. 147–152, 1988.
- [Hastings 1970]** Hastings, W Keith, “Monte Carlo sampling methods using Markov chains and their applications”.
- [He 2015]** K. He, X. Zhang, S. Ren, J. Sun, “Deep Residual Learning for Image Recognition”, *arXiv 1512.03385*, 2015.
- [He 2016]** He, Kaiming, X. Zhang, S. Ren und J. Sun, “Deep residual learning for image recognition”. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, S. 770–778, 2016.
- [Hoffman 2014]** Hoffman, Matthew D. und A. Gelman, “The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo”. *Journal of Machine Learning Research (JMLR)*, 45(1):1593–1623.
- [Huang 2016]** G. Huang, Z. Liu, K. Q. Weinberger, “Densely Connected Convolutional Networks.” 1608.06993, 2016.
- [Kapela 2015]** R. Kapela, P. Sniatała, A. Błoch, and S. A. Atrem, “Asphalt Surfaced Pavement Cracks Detection Based on Histograms of Oriented Gradients,” 2015.
- [Kardan 2017]** N. Kardan, K.O. Stanley, “Mitigating Fooling with Competitive Overcomplete Output Layer Neural Networks,” *Int. Joint Conf. on Neural Networks (IJCNN)*, 2017.

- [**Kendall 2017**] A. Kendall, Y. Gal, “What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?” Advances in Neural Information Processing Systems (NIPS), 2017.
- [**Kiureghian 2009**] Kiureghian, Armen Der und O. Ditlevsen, „Aleatory or epistemic? Does it matter?”. Structural Safety, 22(1):79–86, 2009.
- [**Krizhevsky 2009**] Krizhevsky, Alex, “Learning multiple layers of features from tiny images”. Technischer Bericht, 2009.
- [**Lakshminarayanan 2017**] B. Lakshminarayanan, A. Pritzel, C. Blundell, “Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles,” Advances in Neural Information Processing Systems (NIPS), 2017.
- [**LeCun 2010**] LeCun, Yann, C. Cortes und C. Burges, „MNIST handwritten digit database“. ATT Labs [Online]. Available: <http://yann.lecun.com/exdb/mnist>, 2.
- [**Lembach 2019**] Lembach, Johann Uwe, “Alignment of Image Sequences for the Analysis of Pavement Distress”. Masterarbeit, TU Ilmenau, 2019.
- [**Lembach 2020**] Lembach, J., Stricker, R., Gross, H.-M., „Efficient Implementation of Regional Mutual Information for the Registration of Road Images.“, in: Int. Conf. on Image Processing Theory, Tools and Applications (IPTA), Paris, France, 6 pages, IEEE 2020
- [**Leutenegger 2011**] Leutenegger, Stefan, M. Chli und R. Y. Siegwart, „BRISK Binary Robust Invariant Scalable Keypoints”. International Conference on Computer Vision ICCV 2011.
- [**Lin 2017**] Lin, Tsung-Yi, P. Goyal, R. Girshick, K. He und P. Dollar, “Focal Loss for Dense Object Detection”. In: Proceedings of the IEEE International Conference on Computer Vision, 2017.
- [**Long 2015**] Long, Jonathan, E. Shelhamer und T. Darrell, “Fully Convolutional Networks for Semantic Segmentation”. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), S. 3431—3440, 2015.
- [**Lowe 2004**] Lowe, David G., “Distinctive Image Features from Scale-Invariant Keypoints”. International Journal of Computer Vision, 60(2):91–110, 2004.
- [**Mair 2010**] Mair, Elmar, G. D. Hager, D. Burschka, M. Suppa und G. Hirzinger, “Adaptive and Generic Corner Detection Based on the Accelerated Segment Test”. European Conference on Computer Vision ECCV 2010, S. 183–196, 2010.
- [**Mikolajczyk 2005**] Mikolajczyk, Krystian und C. Schmid, “A Performance Evaluation of Local Descriptors”. IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(10):1615–1630, 2005.
- [**Minka 2004**] Minka, Thomas, “Power EP”, 2004.
- [**Mur-Artal 2015**] R. Mur-Artal, J. M. M. Montiel and J. D. Tardós, "ORB-SLAM: A Versatile and Accurate Monocular SLAM System," in IEEE Transactions on Robotics, vol. 31, no. 5, pp. 1147-1163, 2015.

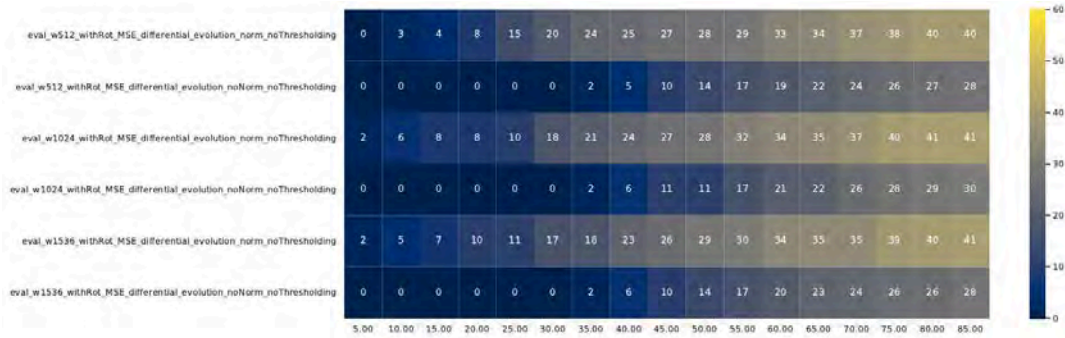
- [Oliveira 2014]** H. Oliveira and P. L. Correia, “CrackIT - An image processing toolbox for crack detection and characterization,” in 2014 IEEE International Conference on Image Processing (ICIP). IEEE, pp. 798–802, 2014.
- [Oliveira 2017]** H. Oliveira and P. L. Correia, “Road surface crack detection: Improved segmentation with pixel-based refinement,” 2017 25th European Signal Processing Conference (EUSIPCO), pp. 2026–2030, 2017.
- [Pauly 2017]** L. Pauly, D. Hogg, R. Fuentes, and H. Peel, “Deeper networks for pavement crack detection,” pp. 479–485, 2017.
- [Peng 2015]** L. Peng, W. Chao, L. Shuangmiao, and F. Baocai, “Research on crack detection method of airport runway based on twice-threshold segmentation,” Int. Conf. on Instrumentation and Measurement, Computer, Communication and Control (IMCCC). IEEE, pp. 1716–1720, 2015.
- [Quintana 2016]** M. Quintana, J. Torres, and J. M. Menendez, “A Simplified Computer Vision System for Road Surface Inspection and Maintenance,” IEEE Transactions on Intelligent Transportation Systems, vol. 17, no. 3, pp. 608–619, 2016.
- [Ronneberger 2015]** Ronneberger, O., P. Fischer und T. Brox, „U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: Medical Image Computing and Computer Assisted Intervention (MIC-CAI), Bd. 9351 d. Reihe LNCS, S. 234–241. Springer. (available on arXiv:1505.04597 [cs.CV]), 2015.
- [Rosten 2006]** Rosten, Edward und T. Drummond (2006). Machine Learning for High-Speed Corner Detection. European Conference on Computer Vision ECCV 2006, S. 430–443.
- [Ruble 2011]** Rublee, Ethan, V. Rabaud, K. Konolige und G. Bradski, “ORB: an efficient alternative to SIFT or SURF”. International Conference on Computer Vision ICCV 2011, S. 2564–2571, 2011.
- [Russakoff 2004]** Russakoff, Daniel B., C. Tomasi, T. Rohlfing und C. R. Maurer (2004). Image similarity using mutual information of regions. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 3023(May 2004):596–607.
- [RVS 11.06.69]** Qualitätssicherung Bau - Prüfungen – Fahrbahnoberfläche. Digitale Hochgeschwindigkeitsbilderfassung der Fahrbahnoberfläche mit dem System RoadSTAR; FSV, Wien, 2009.
- [RVS 13.01.11]** Qualitätssicherung bauliche Erhaltung - Bauliche Straßenerhaltung - Pavement Management. Zustandbeschreibung und mögliche Schadensursachen von Asphalt- und Betonstraßen; FSV, Wien, 2009.
- [RVS 13.01.16]** Qualitätssicherung bauliche Erhaltung Bauliche Straßenerhaltung - Pavement Management. Bewertung von Oberflächenschäden und Rissen auf Asphalt- und Betondecken, FSV, Wien, 2013.

- [Salehi 2017] Salehi, Seyed Sadegh Mohseni, D. Erdogmus und A. Gholipour, „Tversky loss function for image segmentation using 3D fully convolutional deep networks”. Computing Research Repository (CoRR), abs/1706.05721, 2017.
- [Sandler 2018] Sandler, Mark, A. Howard, M. Zhu, A. Zhmoginov und L.-C. Chen, „Mobilenetv2: Inverted residuals and linear bottlenecks”. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), S. 4510–4520, 2018.
- [Schmidt, 2021] Schmidt, Friedrich, „Deep-Learning-basierte Klassifikation von Strassenoberflächen“, Masterarbeit, TU-Ilmenau, 2021.
- [SN 640 925b] Erhaltungsmanagement der Fahrbahnen (EMF) - Zustandserhebung und Indexbewertung, VSS, Zürich, 2003.
- [SN 640 925b Anhang] Erhaltungsmanagement der Fahrbahnen (EMF) - Anleitung zur visuellen Zustandserhebung und Indexbewertung mit dem Schadenkatalog, VSS, Zürich, 2003.
- [Seichter 2018] D. Seichter, M. Eisenbach, R. Stricker, H.-M. Gross, “How to Improve Deep Learning based Pavement Distress Detection while Minimizing Human Effort,” Int. Conf. on Automation Science and Engineering (CASE), 2018.
- [Simonyan 2014] K. Simonyan, A. Zisserman, „Very Deep Convolutional Networks for Large-Scale Image Recognition”, arXiv 1409.1556, 2014.
- [Stephan 2019] Benedict Stephan, „Schätzung der Unsicherheiten in Deep-Learning-basierten Lernverfahren“, Masterarbeit, TU Ilmenau, 2019.
- [Stricker 2021] Stricker, R., Aganian, D., Sesselmann, M., Seichter, D., Engelhardt, M., Spielhofer, R., Hahn, M., Hautz, A., Debes, K., Gross, H.-M., „Road Surface Segmentation - Pixel Perfect Distress and Object Detection for Road Assessment.“ in: IEEE Int. Conf. on Automation Science and Engineering (CASE), pp. 1-8, IEEE, 2021
- [Storn 1997] Storn, Rainer, “Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces”. Journal of Global Optimization, 11:341–359, 1997.
- [Tang 2013] J. Tang and Y. Gu, “Automatic crack detection and segmentation using a hybrid algorithm for road distress analysis,” Proceedings – 2013 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2013, pp. 3026–3030, 2013.
- [Thrun 2005] S. Thrun, W. Burgard, and D. Fox, “Probabilistic Robotics”, MIT Press, Cambridge, MA, 2005.
- [Thrun 2012] S. Thrun, L. Pratt, “Learning to Learn”, Springer Science & Business Media, (ISBN-13: 9780792380474, 2012.
- [Tola 2010] Tola, Engin, V. Lepetit und P. Fua, “DAISY: An Efficient Dense Descriptor Applied to Wide-Baseline Stereo”. IEEE Transactions on Pattern Analysis and Machine Intelligence, 32(5):815–830, 2010.
- [Varadharajan 2014] S. Varadharajan, S. Jose, K. Sharma, L. Wander, and C. Mertz, “Vision for road inspection,” 2014 IEEE Winter Conference on Applications of Computer Vision, WACV 2014, pp. 115–122, 2014.

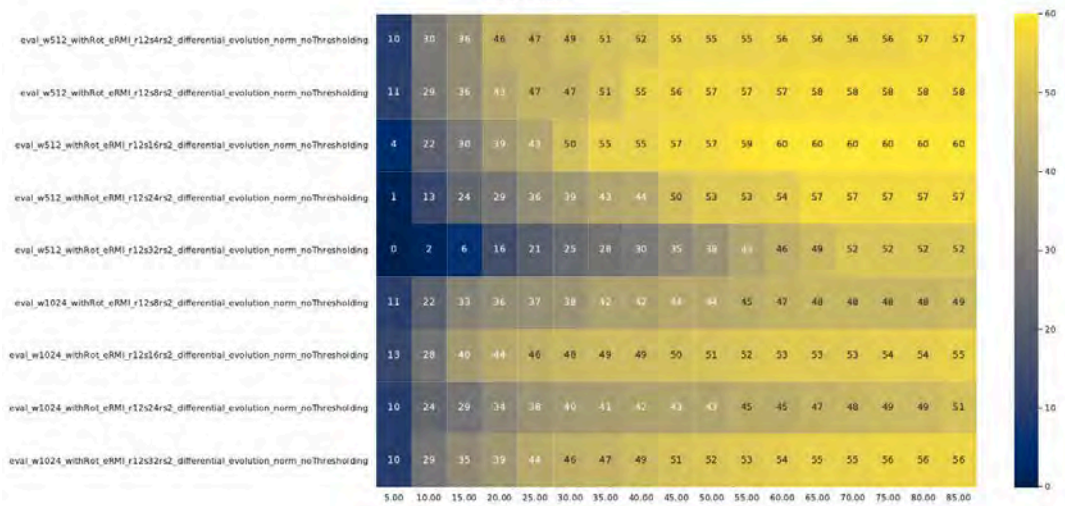
- [Wang 2017]** X. Wang and Z. Hu, “Grid-based pavement crack analysis using deep learning,” in 2017 4th International Conference on Transportation Information and Safety (ICTIS), pp. 917–924, 2017.
- [Xu 2013]** K. Xu, N. Wei, and R. Ma, “Pavement crack image detection algorithm under nonuniform illuminance,” Int. Conf. on Information Science and Technology (ICIST). IEEE, pp. 1281–1284, 2013.
- [Zhao, 2017]** Zhao, Hengshuang, J. Shi, X. Qi, X. Wang und J. Jia (2017). Pyramid scene parsing network. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.
- [Zou 2012]** Q. Zou, Y. Cao, Q. Li, Q. Mao, and S. Wang, “CrackTree: Automatic crack detection from pavement images,” Pattern Recognition Letters, vol. 33, no. 3, pp. 227–238, 2012.
- [ZTV ZEB-StB 06]** Zusätzliche Technische Vertragsbedingungen für die Zustandserfassung und Bewertung von Straßen, 2006, Forschungsgesellschaft für Straßen- und Verkehrswesen.
- [ZTV Funktion Stb, 2004]** Zusätzliche Technische Vertragsbedingungen für Funktionsbauverträge, 2004, Forschungsgesellschaft für Straßen- und Verkehrswesen.
- [ZTV Mt Funktion]** Zusätzliche Technische Vertragsbedingungen für Funktionsbauverträge für die messtechnische Erfassung, 2004, Forschungsgesellschaft für Straßen- und Verkehrswesen.

14 Anhang

14.1 Anhang A

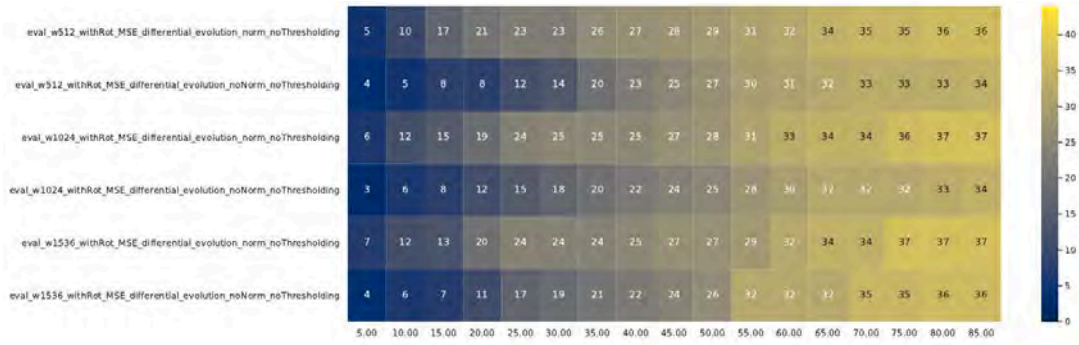


(a) MSE

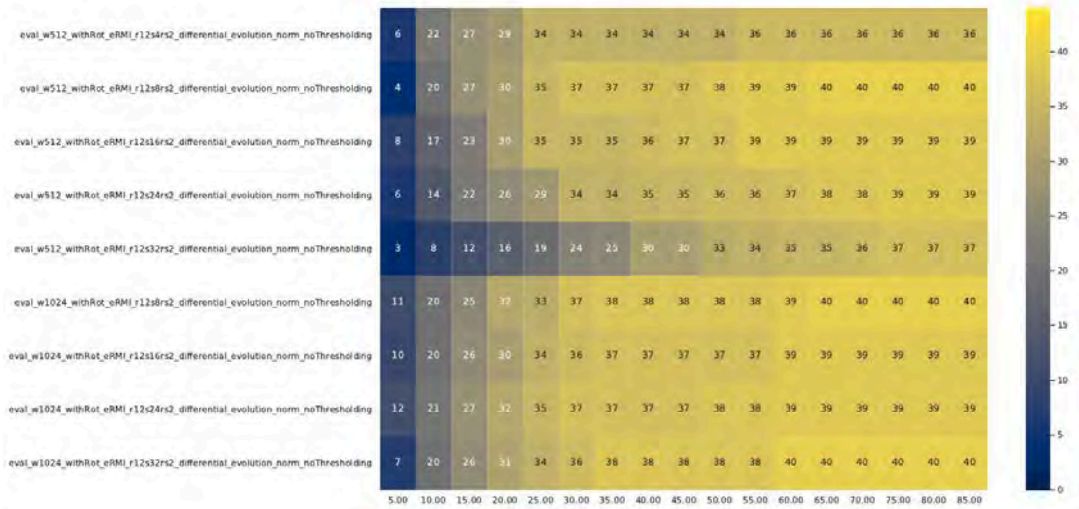


(b) RMI

Abbildung 77: Autobahn (2014-18): Anzahl gefundener Transformationen mit Evaluationsmaß D unter Grenze

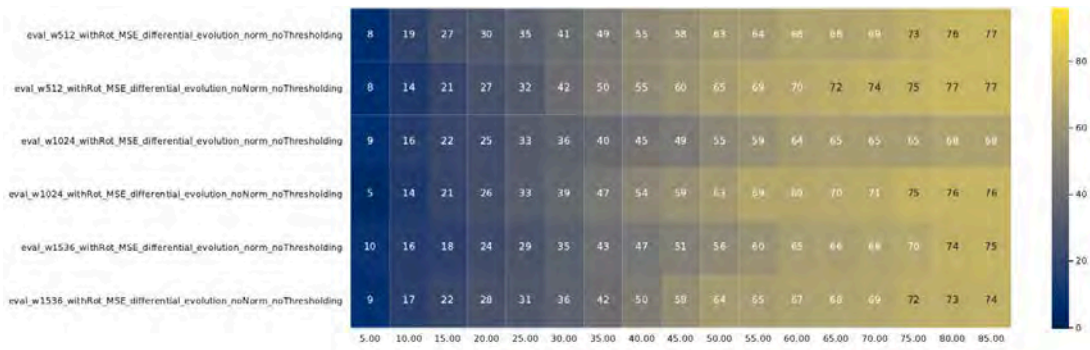


(a) MSE

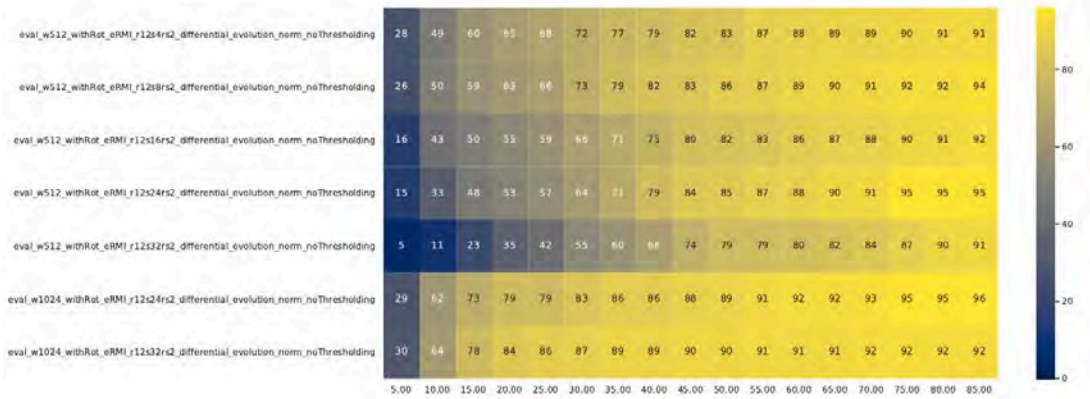


(b) RMI

Abbildung 78: Innerorts (2018-19): Anzahl gefundener Transformationen mit Evaluationsmaß D unter Grenze



(a) MSE



(b) RMI

Abbildung 79: Bundesstraße (2011-15): Anzahl gefundener Transformationen mit Evaluationsmaß D unter Grenze