ARGLOS – AUTOMATIC RECOGNITION OF LEVEL OF SERVICE

In dem Projekt ARGLOS (Automatic RecoGnition of Level Of Service) wurde ein System entwickelt, welches anhand von Bildern der Webcams der ASFiNAG automatisch die aktuelle Verkehrslage ermittelt.

Zusammenfassung

Die ASFiNAG betreibt im hochrangigen Straßennetz zahlreiche Webcams, die über die Website sowie die Unterwegs App der ASFiNAG öffentlich zugänglich sind. Dadurch können sich Verkehrsteilnehmer über die Verkehrssituation auf gewünschten Streckenabschnitten informieren. Zu Beginn des Projekts waren es 550 Webcams, die ASFiNAG baut aber weiter aus und inzwischen sind es bereits 660. Es wurde nun eine Bildverarbeitung entwickelt, die aus den Bildern dieser Webcams automatische die aktuelle Verkehrslage ableitet. Die besondere Herausforderung hierbei war die niedrige Qualität und Auflösung der Bilder, sowie die geringe Bildrate (ein Bild pro Sekunde). Zusätzlich beanspruchen die Algorithmen wenig Rechenleistung, so dass Standard Server verwendet werden können.

Im Rahmen des Projekts wurden alle Webcams der A23 in Wien in das System integriert, das System arbeitet jedoch unabhängig von den Kameras, Verkehrslagen, Witterungsbedingungen sowie bei Tag und bei Nacht. Die Verkehrsinformationen werden dabei über eine einfache http Schnittstelle im XML Format minütlich zur Verfügung gestellt.

Facts:

- Laufzeit: 07/2015-07/2016
- Auftragnehmer: EFKON AG
- Automatische Verkehrslageinformationen
- Auflösung: 352x288
- Alle zwei Sekunden ein Bild
- Update jede Minute
- Erkennrate bei 92,9%
- Server: Intel Xeon E3-1220v3, 32GB RAM
- Bis zu 315 Kameras parallel

ABB 1. Freie, dichte und gestaute Verkehrslage

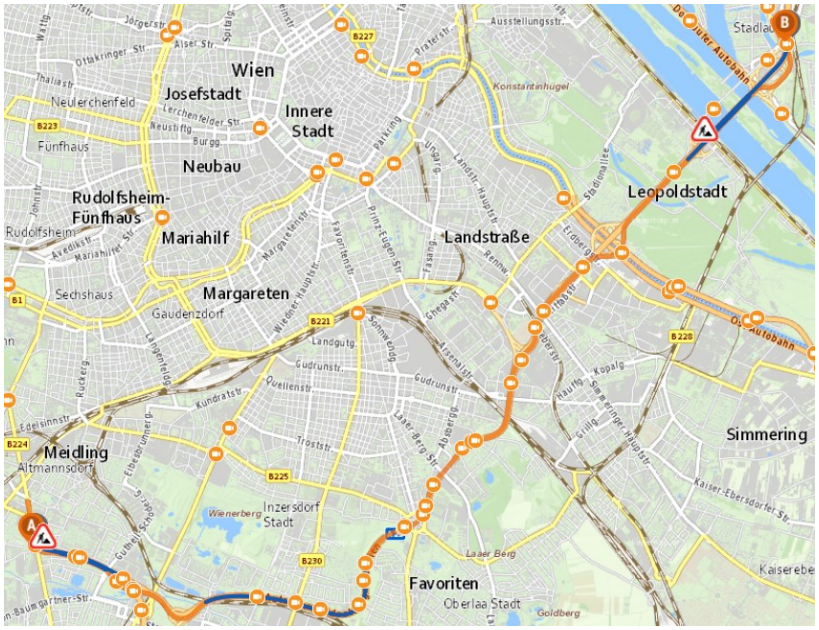

ABB 2. Überwachte Kameras des Projekts ARGLOS

Kurzzusammenfassung

Problem
Ziel des Projekts war es, Verkehrslageinformationen auf den hochrangigen Straßen durch die bereits vorhandenen ca. 550 Webcams automatisch zu erheben. Besonders herausfordernd war die niedrige Qualität und Auflösung der Bilder, sowie die geringe Bildrate. Zusätzlich sollte wenig Rechenleistung beansprucht werden.

Gewählte Methodik
Zwei Ansätze wurden implementiert: Ein sehr einfacher Median-Algorithmus, der äußerst schnell zu rechnen ist, der aber vor allem bei freien Verkehrslagen gut arbeitet und ein deutlich komplexerer, aber in allen Lagen performanter Algorithmus auf Basis von Convolutional Neural Networks, der zugeschaltet wird, wenn der erste Algorithmus keinen freien Verkehr entdeckt.

Ergebnisse
Auf einer Stichprobe mit schwierigen Verkehrsverhältnissen hat der Ansatz eine Erkennrate von 92,9% erzielt. Die Daten wurden auf allen WebCams der A23 in Wien erhoben.

Schlussfolgerungen
Eine automatische Auswertung von Bildern der Webcams ist mit heutigen Bildverarbeitungsmethoden mit ausreichender Genauigkeit möglich. Dadurch können neue Streckenabschnitte schnell für Verkehrslageinformationen verfügbar gemacht werden.

English Abstract

The objective of this project is to automatically deduce traffic congestion information from the currently installed 550 webcams on the major road network. A particular challenge were the low quality and resolution of the images as well as the low image rate. In addition, the speed of the algorithms was important.

Impressum:

Bundesministerium für Verkehr, Innovation und Technologie
DI Dr. Johann Horvatits,
Abt. IV/ST 2 Technik und Verkehrssicherheit
johann.horvatits@bmvit.gv.at,

DI (FH) Andreas Blust,
Abt. III/I4 Mobilitäts- und Verkehrstechnologien
andreas.blust@bmvit.gv.at,
www.bmvit.gv.at

ÖBB-Infrastruktur AG
Ing. Wolfgang Zottl, ISM;
Leitung Forschung & Entwicklung
wolfgang.zottl@oebb.at,
www.oebb.at

ASFINAG
DI Eva Hackl,
Manager International Relations und Innovation
eva.hackl@asfinag.at,

DI (FH) René Moser, Leiter Strategie, Internationales und Innovation
rene.moser@asfinag.at,
www.asfinag.at

Österreichische Forschungsförderungsgesellschaft mbH
DI Dr. Christian Pecharda,
Programmleitung Mobilität
Sensengasse 1, 1090 Wien
christian.pecharda@ffg.at,
www.ffg.at

August, 2016

Automatic RecoGnition of Level Of Service
ARGLOS

Ein Projekt finanziert im Rahmen der Verkehrsinfrastrukturforschung 2014 (**VIF2014**)

August 2016

Impressum:

Herausgeber und Programmverantwortung:
Bundesministerium für Verkehr, Innovation und Technologie
Abteilung Mobilitäts- und Verkehrstechnologien
Radetzkystraße 2
A - 1030 Wien

ÖBB-Infrastruktur AG
Praterstern 3
A - 1020 Wien

Autobahnen- und Schnellstraßen-Finanzierungs
Aktiengesellschaft
Rotenturmstraße 5-9
A - 1010 Wien

Für den Inhalt verantwortlich:
EFKON AG
Dietrich-Keller-Str. 20
8074 Raaba

Programmmanagemen:
Österreichische Forschungsförderungsgesellschaft mbH
Bereich Thematische Programme
Sensengasse 1
A – 1090 Wien

Automatic RecoGnition of Level of Service

ARGLOS

Ein Projekt finanziert im Rahmen der Verkehrsinfrastrukturforschung (VIF2014)

AutorInnen:

Dr. Marcus E. HENNECKE

Auftraggeber:
Bundesministerium für Verkehr, Innovation und Technologie
ÖBB-Infrastruktur AG
Autobahnen- und Schnellstraßen-Finanzierungs-Aktiengesellschaft

Auftragnehmer:
EFKON AG

INHALTSVERZEICHNIS

1 EXECUTIVE SUMMARY

Derzeit werden Verkehrslageinformationen auf den hochrangigen Straßen durch die ASFiNAG mit Hilfe von Sensoren und aus dem GO-Maut System erhoben. Entlang der Straßen sollten zukünftig auch die bereits vorhandenen ca. 550 Webcams eingesetzt werden. Hierzu war eine entsprechende automatische Bildverarbeitung und –interpretation zu entwickeln, die anhand des Bildmaterials die aktuelle Verkehrslage abschätzt. Es sollte mindestens zwischen freiem, dichtem und gestautem Verkehr unterschieden werden. Die besondere Herausforderung hierbei war die niedrige Qualität und Auflösung der Bilder, sowie die geringe Bildrate (ein Bild pro Sekunde). Außerdem sollten die Algorithmen möglichst wenig Rechenleistung beanspruchen.

Die Fusion der aus den Kameras generierten Verkehrslageinformationen mit den Informationen der bereits bestehenden Systeme war nicht Ziel des Projekts.

Alle Ziele des Projekts wurden vollinhaltlich erreicht. Es wurde ein System entwickelt und im Dauerbetrieb zur Verfügung gestellt, welches minütlich für alle Kameras der A23 für jede Fahrtrichtung Verkehrslageinformationen generiert und dabei sieben verschiedene Klassen von frei bis Stillstand unterscheidet. Dieses System läuft auf einem Rechner und verwendet nur einen von vier Prozessorkernen.

Auf einem Datensatz mit einem hohen Anteil von Stausequenzen wurde eine Klassifikationsgenauigkeit von 92,9% gemessen.

Eine wichtige Erkenntnis war, dass die Kameras zwar einen Streckenabschnitt überwachen, im Verhältnis zur gesamten Länge einer Autobahn jedoch als punktuelle Sensoren angesehen werden müssen. Selbst in einem dichten Stau gibt es immer wieder Bereiche in denen der Verkehr frei fließt. Daher können die aus den Kameras generierten Informationen schwanken und von den Informationen der anderen Systeme abweichen.

Da das Projekt aufgrund vertraglicher Schwierigkeiten nicht zum 1.7.2015 starten konnte, wurde das Projektende auf den 31.7.2016 verschoben.

2 ARBEITSPAKETE UND MEILENSTEINE

2.1 Übersichtstabellen

Tabelle 1: Arbeitspakete

AP Nr.	Arbeitspaket Bezeichnung	Fertigstellungsgrad	Basistermin		Aktuell		Erreichte Ergebnisse / Abweichungen
			Anf.	Ende	Anf.	Ende	
1	Projektmanagement	100%	07/15	06/16	07/15	07/16	Abgeschlossenes Projekt
2	Algorithmen-entwicklung	100%	08/15	05/16	08/15	06/16	Algorithmen zur Bestimmung der Verkehrslage
3	Systementwicklung	100%	07/15	12/15	08/15	03/16	Lauffähiges System mit Schnittstellen zu den ASFiNAG Systemen
4	Datensammlung	100%	10/15	04/16	10/15	06/16	Ca. 1,3 Millionen Patches, 8000 annotierte Sequenzen
5	Evaluation	100%	11/15	05/16	12/15	07/16	Kontinuierliche Evaluation der entwickelten Algorithmen

Tabelle 2: Meilensteine

Meilen-stein Nr.	Meilenstein Bezeichnung	Basis-termin	Akt. Planung	Meilenstein erreicht am	Anmerkungen zu Abweichungen
1	Schnittstellenspezifikation	17.08.15	06.08.15	06.08.15	
2	System bereit f. Aufzeich-nungen	01.10.15	01.10.15	01.10.15	Aufzeichnungen wurden ge-startet; System mit Verkehrs-lageinformationen erstmals mit 31.03.16 erreichbar
3	Zwischenbericht	31.12.15	31.12.15	31.12.15	
4	Abschlussbericht	30.06.16	19.08.16	19.08.16	Abschlusspräsentation wurde am 09.08.16 gehalten

2.2 Beschreibung der im Berichtszeitraum durchgeführten Arbeiten

In diesem Kapitel werden die im Projekt insgesamt durchgeführten Arbeiten nach Arbeitspaket gegliedert dargestellt. Die bei der Abschlusspräsentation vorgestellten PowerPoint Folien enthalten mehrere Videos zur Veranschaulichung der Sachverhalte und sie stellen daher einen Anhang zu diesem Bericht dar.

2.2.1 AP1 Projektmanagement

Das Projekt wurde mit Unterzeichnung des Finanzierungsvertrags am 14. Juli 2015 offiziell gestartet. Damit wurden auch die entsprechenden Kostenstellen eingerichtet und die ersten Arbeitspakete gestartet. Das Kickoff mit der EFKON und der ASFiNAG fand am 5. August 2015 in Raaba bei Graz statt.

Die Berichte für die Zeiträume Juli/August und September/Oktober sowie der Zwischenbericht wurden erstellt. Ab dem zweiten Halbjahr wurden monatliche

Statusmeetings durchgeführt um die jeweils aktuellen Entwicklungen vorzustellen. Hinzu kamen nach Bedarf weitere Besprechungen, teils auch über Skype.

2.2.2 AP2 Algorithmenentwicklung

Herausforderungen

Aufgrund der geringen Auflösung und der geringen Framerate war rasch klar, dass der typische Detect-and-track Ansatz nicht funktionieren würde, siehe Abbildung 1. Bei diesem Ansatz werden in einem festgelegten Bereich die Fahrzeuge detektiert und dann über den gesamten Abschnitt verfolgt. Voraussetzung dafür ist allerdings, dass jedes Fahrzeuge in mehreren Bildern sichtbar ist und sich dabei ähnlich sieht.

Abbildung 1: LKW (rot markiert) in aufeinander folgenden Bildern

Eine weitere Herausforderung stellten ungünstige Lichtverhältnisse und Niederschläge dar, teilweise auch in Kombination.

Abbildung 2: Gegenlicht (links) und starker Niederschlag (rechts)

Daher wurden verschiedene Algorithmen untersucht, die stattdessen einen kompletten Bereich überwachen und darin auf direkte Art und Weise die Verkehrslage schätzen. Dazu zählen texturbasierte Verfahren ebenso wie Ansätze des Optical Flow oder auch Klassifikationsansätze. Letztlich erfolgreich war eine Kombination aus einem sehr einfachen, aber schnell zu rechnenden Median Ansatz und einem komplexeren und zuverlässigeren, aber langsameren Ansatz basierend auf Neuronalen Netzen.

Region of Interest

Um für unterschiedliche Fahrtrichtungen jeweils unterschiedliche Verkehrslageinformationen schätzen zu können, müssen jeder Kamera ein oder mehrere Regions of Interest (ROI) zugeordnet werden, siehe Abbildung 3. Diese können eine Hauptfahrbahn oder auch eine Auf- oder einer Abfahrt umschließen. Für jede ROI wird die Verkehrslage separat geschätzt.

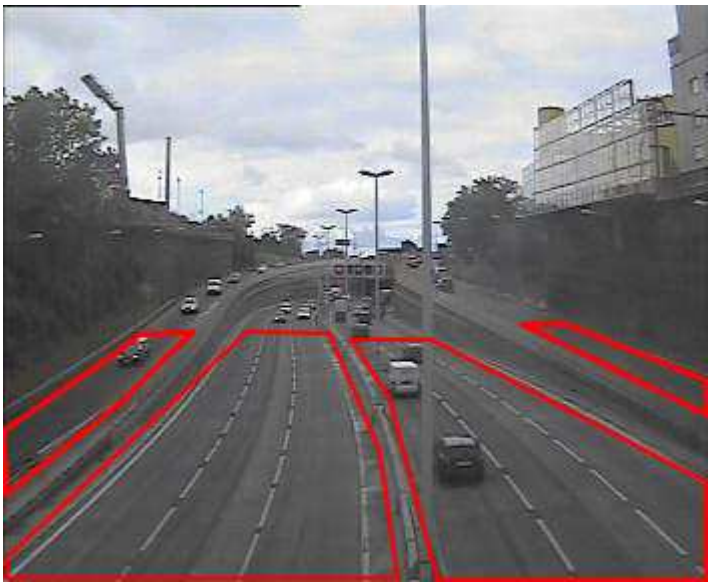

Abbildung 3: Region of Interest (ROI)

Median Ansatz

Die Grundidee des Median Ansatzes ist es, aus den Bildern einer Sequenz zunächst ein Bild der Fahrbahn ohne Fahrzeuge zu erzeugen und dieses dann mit allen Bildern der Sequenz zu vergleichen. Dort wo es Abweichungen gibt, muss sich ein Fahrzeug befinden und damit kann die mittlere Verkehrsdichte abgeschätzt werden.

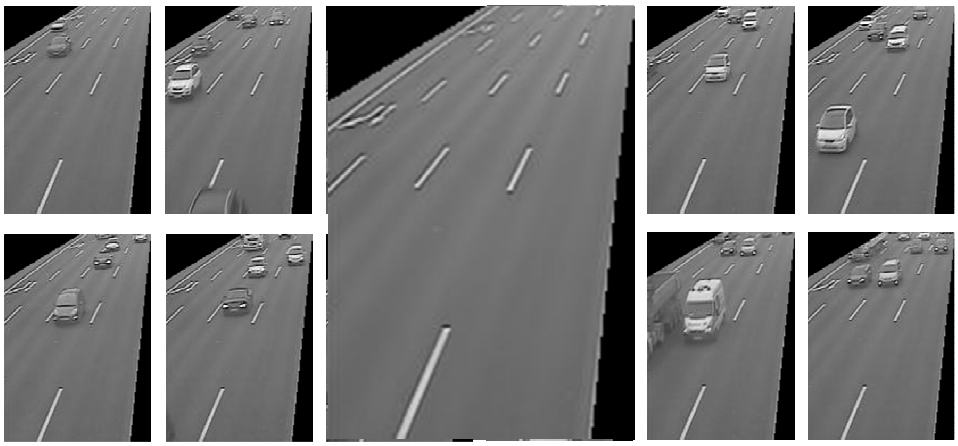

Abbildung 4: Erzeugung des Medianbildes

Abbildung 4 zeigt, wie das Bild der Fahrbahn erzeugt wird. Aus den Bildern der Sequenz (links und rechts) wird ein Medianbild gerechnet. Das heißt, dass für jedes Pixel der Medianwert über alle Bilder der Sequenz bestimmt wird. Solange das Pixel in mindestens 50% der Bilder nicht von einem Fahrzeug bedeckt ist, findet der Median den richtigen Grauwert für die Fahrbahn.

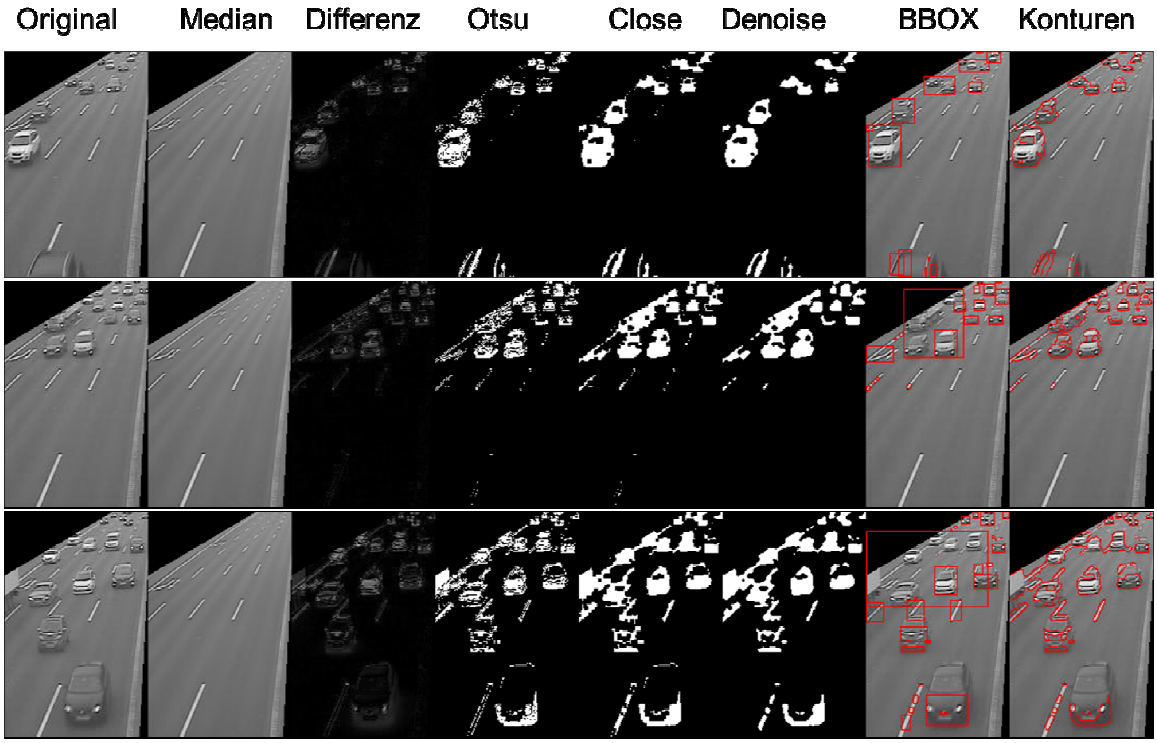

Abbildung 5: Bestimmung der Verkehrsdichte aus dem Medianbild

Abbildung 5 zeigt die weiteren Schritte: Von jedem Bild der Sequenz (Original) wird das Medianbild abgezogen. Der Betrag der Differenz ergibt dann das Differenzbild. Aus jedem Differenzbild dann mit dem Otsu Ansatz ein adaptiver Schwellwert errechnet, mit dem das

Differenzbild binarisiert wird. Alle Pixel, deren Grauwert über dem Schwellwert liegen, werden auf weiß gesetzt, alle anderen auf Schwarz. Dieses Binärbild wird dann noch einer Close und einer Denoise Operation unterzogen um zu einem Bild zu kommen, welches die Fahrzeuge von der Fahrbahn unterscheidet. Zwei Varianten wurden dann in Folge untersucht, einmal die Verwendung von Bounding Boxes (BBOX) und einmal von Konturen. Pixel, die sich jeweils innerhalb der Boxen bzw. der Konturen befanden, wurden als Fahrzeugpixel gezählt. Das Verhältnis aus diesen Pixeln zu der Gesamtzahl der Pixel wurde als Maß der Verkehrsdichte angenommen. Es zeigte sich, dass Konturen in diesem Fall besser funktionieren.

Auf einer repräsentativen Stichprobe, die über 24 Stunden eines Tages aufgezeichnet wurde, zeigte der Medianansatz eine ausgezeichnete Genauigkeit, siehe Kapitel 2.2.5. Es stellte sich allerdings heraus, dass diese Genauigkeit täuscht. Der Medianansatz funktioniert sehr gut, solange die Verkehrsdichte unter 50% bleibt. Bei dichtem bis gestautem Verkehr bricht die Genauigkeit jedoch stark ein. Der Grund für die anscheinend hohe Genauigkeit liegt darin, dass der Verkehr in den weitaus meisten Fällen frei bzw. frei bis dicht ist. Verkehrsstaus kommen vergleichsweise selten vor. Daher musste ein zweiter Algorithmus entwickelt werden, welcher in Stausituationen zum Einsatz kommt und daher auch eine höhere Komplexität aufweisen darf.

Neuronale Netze

Ein sehr performanter Ansatz in der Bilderkennung sind die so genannten Convolutional Neural Networks. Dabei sind die einzelnen Neuronen derart angeordnet, dass sie auf sich überlappende Bereiche reagieren. Mathematisch kann man diesen Prozess als diskrete Faltung interpretieren, was den Zusatz convolutional erklärt. Damit stellen CNNs eine Sonderform von mehrlagigen Perzeptrons dar.

Seit dem Einsatz von Grafikprozessor-Programmierung können CNNs erstmals effizient trainiert werden und gelten als state of the art Methode für zahlreiche Anwendungen im Bereich der Klassifizierung und Erkennung, wie etwa Gesichtserkennung, Bilderkennung oder Spracherkennung. Diese Problemstellungen erfordern häufig Architekturen, die aus einer Vielzahl von convolutional layers (etwa Faltungs-Schichten) bestehen. Oft werden diese Netzwerke dann als deep convolutional neural networks (DCNNs) bezeichnet und fallen unter die Disziplin Deep Learning. (Wikipedia)

Abbildung 6 zeigt ein Beispiel für ein CNN aus dem Bereich der Verkehrsschilderkennung. Es handelt sich um ein Neuronales Netz mit sehr vielen Schichten. Die Interpretation der ersten Schichten geht dabei dahin, dass es sich hierbei um eine Merkmalsextraktion handelt,

die den letzten Schichten die Merkmale für die eigentliche Klassifikation liefert. Trainiert wird aber das komplette Netzwerk, so dass nicht nur der Klassifikator sondern auch die Merkmalsextraktion der Problemstellung passend trainiert wird. Der Vorteil dieser Interpretation ist, dass man die Merkmalsextraktion separat trainieren kann. Ein Nachteil der vielschichtigen Netze ist es, dass aufgrund der vielen zu trainierenden Gewichte auch ein sehr großer Datensatz benötigt wird. Diesen vollständig zu annotieren ist jedoch sehr aufwändig und fehleranfällig und übersteigt häufig die Mittel. Die Merkmalsextraktion kann jedoch mit Hilfe eines Autoencoders mit nicht annotierten Daten trainiert werden. Anschließend müssen dann nur noch die Schichten des Klassifikators trainiert werden, wozu wesentlich weniger Daten vonnöten sind.

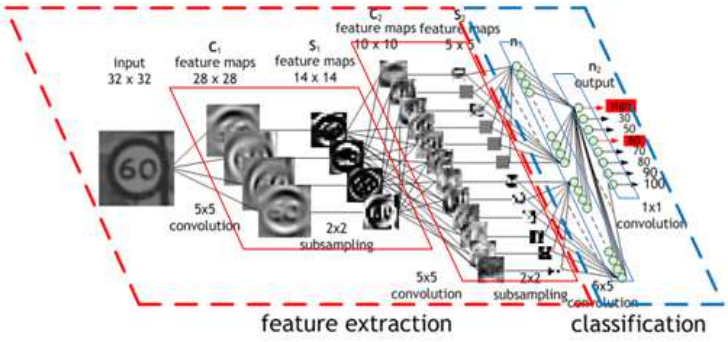

Abbildung 6: Beispiel für Convolutional Neural Networks

In diesem Projekt wurde die Merkmalsextraktion mit 1,3 Millionen nicht annotierten Daten trainiert, der Klassifikator dann mit 7732 manuell annotierten Bildern.

Ein neuronales Netz überblickt immer einen festen Bildausschnitt (Patch). Im Rahmen von ARGLOS wurde dazu eine Patchgröße von 64x64 Pixeln gewählt. Eine Region of Interest variiert jedoch von Kamera sowohl in Größe als auch in Form. Auch eine Sequenz kann in der Länge variieren. Um zu einer Bewertung der ROI und letztlich der Sequenz zu kommen, wurden mehrere, sich überlappende Patches über die ROI gelegt, siehe Abbildung 7 links. Außerdem wird ein Patch ergänzt um den gleichen Patch des vorherigen und des nächsten Bildes, damit auch Bewegungsinformationen verwendet werden können (rechts).

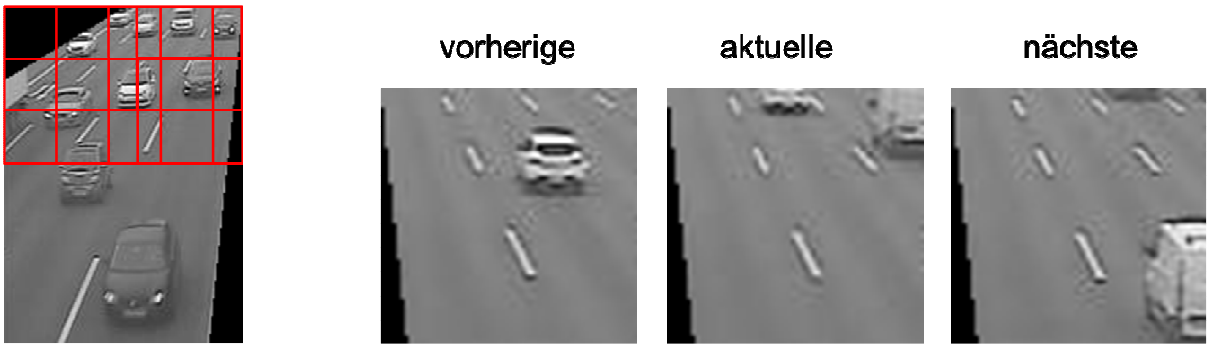

Abbildung 7: Patches werden über die ROI gelegt (links), Patches dreier Bilder werden zusammengefasst (rechts)

Die Eingangsdaten des neuronalen Netzes bestehen also aus drei 64x64 Pixeln großen Patches, insgesamt also 12288 Werten. Das neuronale Netz wurde dann für jeden Patch aufgerufen.

Um das Netz zu trainieren mussten große Mengen an Patches annotiert werden. Insgesamt wurden 7732 Patches annotiert. Die Methode wird in Kapitel 2.2.4 näher erläutert. Das Ergebnis war eine Unterscheidung von 11 Klassen (0 bis 10), die das Netz gelernt hat. Das Netzwerk errechnete dabei für jede der 11 Klassen einen Score, welcher zwischen 0 und 1 liegt. Die Klasse mit dem höchsten Score gewann dann.

Bei beispielsweise 20 Patches in einer ROI und 30 Bildern in der Sequenz führte dies zu insgesamt 600 Patchbewertungen, die anschließend zu einer Gesamtbewertung für die Sequenz zusammengefasst werden mussten. Hierfür wurden zwei Methoden untersucht: Die Mittelwertmethode und wieder ein neuronales Netz.

Für die Mittelwertmethode wurde einfach der Mittelwert über alle Bewertungen gebildet. Auch dieser lag wieder zwischen 0 und 10. Diese Werte mussten über eine Tabelle auf die endgültigen Klassen abgebildet werden.

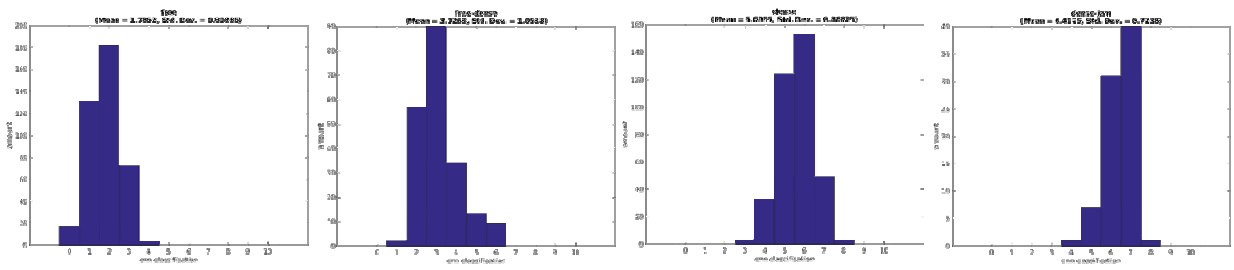

Abbildung 8: Verteilung der Mittelwerte für die verschiedenen Klassen

Dazu wurde für jede der Klassen die Verteilung der Mittelwerte gebildet um daraus dann die Schwellwerte zwischen den Klassen zu bestimmen, siehe Abbildung 8.

Es zeigte sich jedoch, dass der Mittelwert die Klassen zu sehr verwischt und zu keiner scharfen Trennung führt. Die Genauigkeit auf einem Datensatz mit freien und gestauten Sequenzen war nur wenig besser als der Medianansatz.

Daher wurden wieder neuronale Netze zur Klassifikation eingesetzt. Da die Anzahl der Patchbewertungen variierte, musste wieder eine Methode verwendet werden, die zu einer konstanten Größe des Eingangsvektors führt. Dazu wurden die Scores der Klassen, die das CNN lieferte in Histogramme eingeteilt, siehe Abbildung 9. Für jede der elf Klassen wurde dabei ein Histogramm über die Scores mit 10 Bins gebildet. In dem Beispiel hat die Klasse 0 häufig einen Score zwischen 0,2 und 0,3 erhalten, weniger häufig einen Score zwischen 0,1 und 0,2 und die anderen Scores in etwas gleich häufig. Die Klasse 1 hatte ebenfalls eher niedrige Scores, typischerweise zwischen 0,1 und 0,2. Die Klasse 2 hingegen wurde häufig mit hohen Scores zwischen 0,9 und 1 sowie zwischen 0,8 und 0,9 bewertet, während Klasse 3 äußerst niedrige Scores zwischen 0 und 0,1 bekam. Insgesamt erhält man also einen Merkmalsvektor mit 10x11 also 110 Werten.

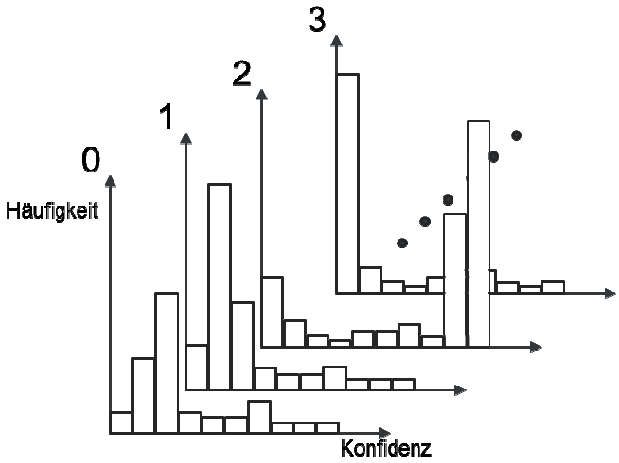

Abbildung 9: Zusammenfassen der Klassenhistogramme zu einem Eingangsvektor

Das neuronale Netz wurde dann auf einem annotierten Trainingsdatensatz trainiert und auf dem unabhängigen Evaluierungsdatensatz getestet. Dieser Ansatz führte zu den besten Ergebnissen von knapp 93%.

2.2.3 AP3 Systementwicklung

Zwei Server wurden ausgewählt und beschafft. Es handelt sich dabei um HP ProLiant DL320e Server mit einem Intel Xeon E3-1220v3 Prozessor und 32GB RAM. Auf diesen

Servern liefen die Entwicklung und die Tests. Mit der ASFiNAG wurde während des Kickoffs vereinbart, dass beide Server bei der EFKON verbleiben und über spezifizierte Schnittstellen mit den ASFiNAG Systemen kommunizieren. Eine Installation bei der ASFiNAG ist dafür nicht erforderlich und seitens ASFiNAG auch nicht erwünscht.

Zum Abholen der Bilder der Web Cams stellte die ASFiNAG ein Web Interface zur Verfügung. Diese relative einfache Schnittstelle basiert auf dem http Protokoll und erlaubt es, zu jeder Kamera das aktuellste Bild abzufragen. Auch für die Übermittlung der Verkehrslage-informationen wurde eine Schnittstelle spezifiziert. Diese werden im XML Format ebenfalls per http übertragen.

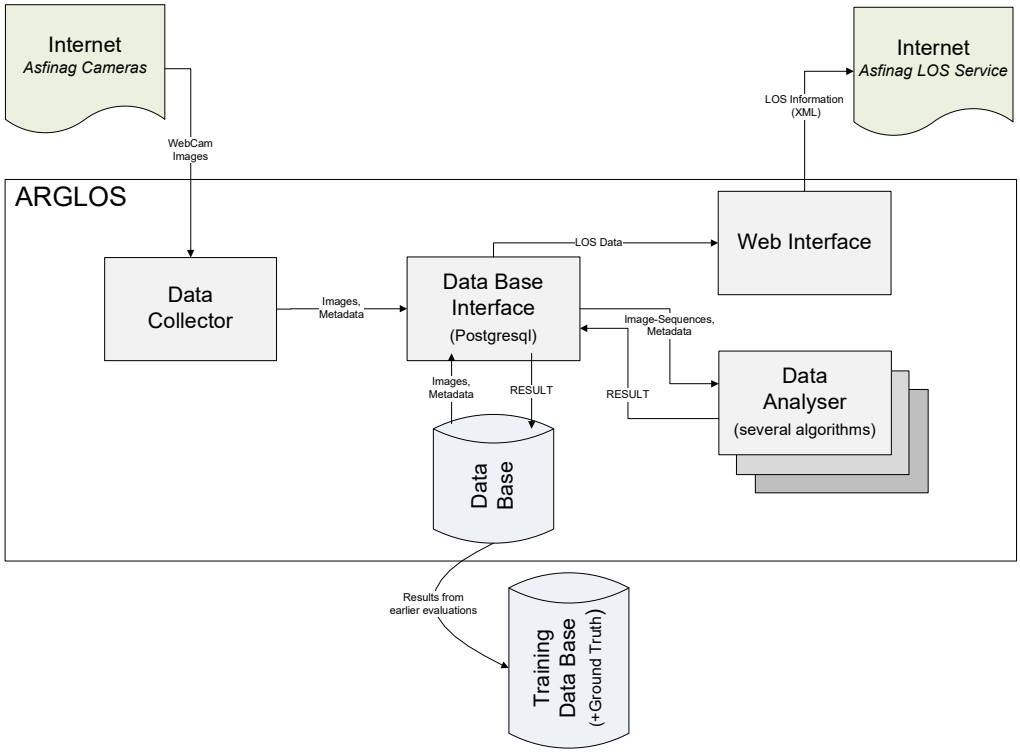

Abbildung 10: Systemarchitektur

Abbildung 10 zeigt das Zusammenspiel aus Data Collector, dem Analyse Modul und der Systemdatenbank bzw. der Trainingsdatenbank. Der Data Collector holt sich in regelmäßigen Abständen Bilder von den Web Cams, speichert sich in der Datenbank und fasst sie zu Sequenzen zusammen. Der Data Analyser wiederum verarbeitet die Sequenzen und schätzt für jede Sequenz die Verkehrslage ab. Über das Web Interface werden die Informationen zur Verfügung gestellt. Das Web Interface kann auch mit einem normalen Browser aufgerufen werden und stellt dann eine umfangreiche GUI zur Verfügung, über die auch historische Daten abgefragt und Bildsequenzen betrachtet werden können.

Die Algorithmen zur Erkennung der Verkehrslage arbeiten mit Bildsequenzen. Es wird immer von jeder Kamera eine Sequenz von beispielsweise 30 Bildern gesammelt, dann werden die Sequenzen bewertet. Bei 660 Web Cams kann es allerdings im zukünftigen Vollbetrieb zu Bandbreitenproblemen kommen. Der Data Collector holt daher nicht von allen Web Cams im Dauerbetrieb Daten sondern immer nur jeweils von einer Gruppe Web Cams. Abbildung 11 zeigt ein Beispiel mit drei Gruppen zu jeweils drei Kameras. In der Abbildung werden von jeder Web Cam Bildsequenzen von 10 Bildern (entsprechend 10 Sekunden langen Bildsequenzen) heruntergeladen bevor dann zur nächsten Gruppe gewechselt wird. Dies geht dann reihum bis von allen Web Cams Bildsequenzen aufgezeichnet worden sind. Auf diese Weise wird in diesem Fall jede Kamera alle 30 Sekunden abgefragt. Diese Bildsequenzen werden in der Datenbank abgelegt und vom Data Analyzer bewertet. Die Extraktion der Verkehrslageinformation beruht dann immer auf diesen Sequenzen.

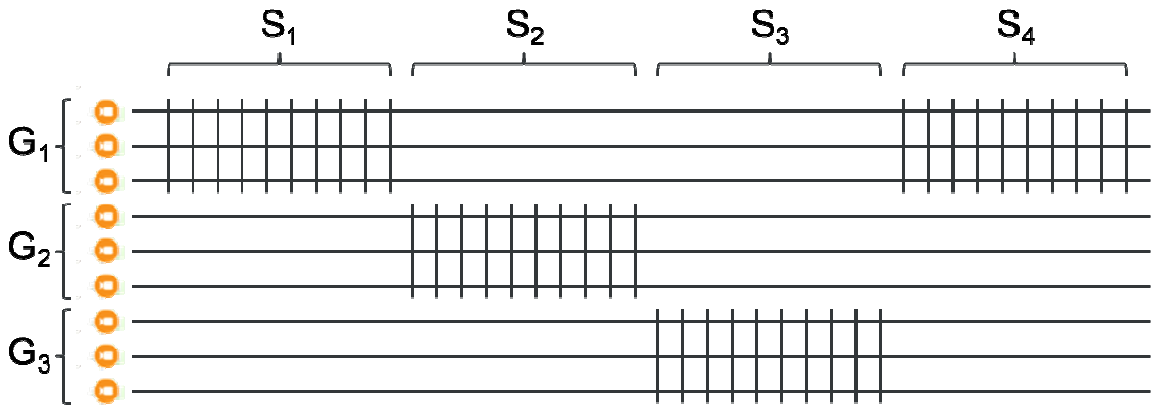

Abbildung 11: Sammeln von Bildsequenzen

Der Vorteil dieser Vorgehensweise liegt vor allem in der geforderten Skalierbarkeit des Systems. Man kann das System auf einem einzelnen, relativ schwachen Rechner kostengünstig laufen lassen, erhält dann aber entsprechend seltener Verkehrslageinformationen. Oder es wird ein leistungsstarker Rechner zur Verfügung gestellt, mit dem sich die Informationen häufiger generieren lassen. Alternativ können die Module leicht auch auf mehrere Rechner verteilt werden. Auch können zum Beispiel der Data Collector oder der Data Analyser auf mehr als einem Rechner laufen. Jede Instanz beschränkt sich dann auf eine Untermenge von Web Cams. Dennoch kann die Datenbank von allen Instanzen benutzt werden, so dass auch kameraübergreifende Informationen verfügbar sind.

Im Projekt wurden alle 15 Kameras der A23 abgefragt, wobei jeder Kamera zwei ROIs zugeordnet waren. Da die Bandbreite dies zuließ, waren alle Kameras in einer Gruppe

zusammengefasst, sie wurden also immer parallel abgefragt. Die Sequenzen enthielten 30 Bilder und es wurde alle zwei Sekunden ein neues Bild geholt. Eine Sequenz entsprach somit einer Minute Aufzeichnungen. Entsprechend wurde jede Minute für alle Kameras die Verkehrslage bewertet.

2.2.4 AP4 Datensammlung

Da relativ frühzeitig ein System zur Verfügung stand, welches automatisiert Daten von den Web Cams sammelt, schien das eigentliche Sammeln der Daten eine sehr einfache Sache zu sein: Aufzeichnungen einschalten und nach der gewünschten Zeit oder Datenmenge wieder abschalten. Anschließend die Daten in die Trainingsdatenbank kopieren (siehe Abbildung 10). Allerdings führte dies zu Datensätzen, die kaum Stausequenzen enthielten. Insbesondere der Medianansatz wurde dadurch deutlich zu optimistisch bewertet, da er gerade bei freien und frei bis dichten Sequenzen sehr gut arbeitet, bei dichteren Situationen jedoch stark abfällt. Auch für das Training der neuronalen Netze war das Ungleichgewicht bei den Klassen nachteilig. Es mussten gezielt wesentlich mehr Stausequenzen aufgezeichnet werden.

Besonders hilfreich erwies sich dabei die Unterwegs App der ASFiNAG, siehe Abbildung 12.

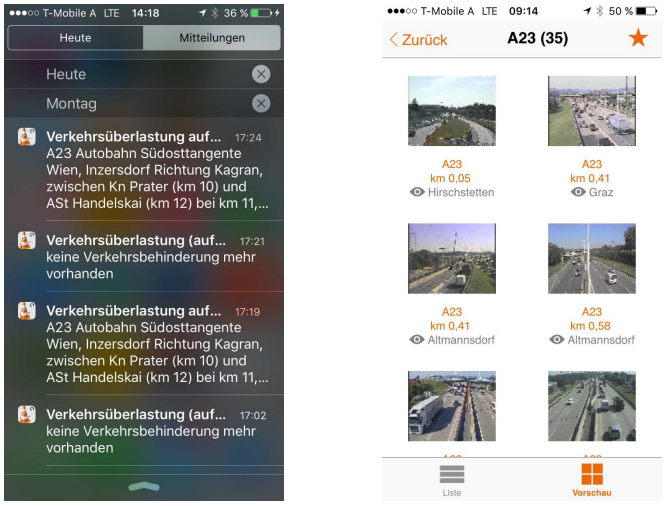

Abbildung 12: Unterwegs App der ASFiNAG

In der App können bestimmte Teilstücke der Autobahnen überwacht werden, so dass automatisch Meldungen geschickt werden, wenn sich auf diesen Teilstücken auffällige Verkehrssituationen ergeben. Da das System kontinuierlich aufzeichnet und die Daten erst nach einer Woche löscht, konnten auf diese Weise viele Stausequenzen gefunden und in die Trainingsdatenbank kopiert werden.

Für das Training und die Evaluierungen wurden die Daten von verschiedensten Kameras aufgezeichnet. Dazu gehören Daten von Kameras in der Grazer Umgebung aber auch alle Kameras der A23 in Wien. Es wurde sorgfältig darauf geachtet, dass zwischen den Datensätzen des Trainings und der Evaluierungen keine Überlappungen bestehen.

Beim Annotieren der Sequenzen ergaben sich allerdings weitere Schwierigkeiten. Jede Sequenz sollte einer der Klassen frei, dicht oder gestaut zugeordnet werden. Die Verkehrslage ist allerdings ein Kontinuum ohne klare Grenzen zwischen den Klassen.

Abbildung 13: Klassen mit Zwischenklassen

Um das System in diesem Kontinuum evaluieren zu können und zu aussagekräftigen Ergebnissen zu kommen wurden zwischen den Klassen noch weitere Klassen eingefügt, siehe Abbildung 13. Es besteht eine starke Überlappung zwischen den Klassen, was die Ambiguität der Klassendefinitionen widerspiegelt. Zusätzlich wurde wegen des Medianansatzes noch die Klasse „Stillstand“ eingeführt. Wenn der Verkehr über die Sequenz hinweg still steht, enthält das Medianbild auch die Fahrzeuge und die Differenzbilder sind nahe Null. Somit würde der Medianansatz in diesem Fall auf frei entscheiden. Es zeigte sich allerdings, dass diese Situation extrem selten vorkommt. Für das Training musste der Fall daher simuliert werden.

Die Convolutional Neural Networks liefern keine direkte Klasse im Sinne von frei-dicht-gestaut sondern schätzen zunächst eine Patch Klasse ab. Es zeigte sich, dass die Patch Klasse nicht aus der Klasse der Sequenz abzuleiten ist. Selbst in dichtem Verkehr kommen Patches vor, die zufälligerweise überhaupt keine Fahrzeuge enthalten. Wenn solche freien Patches als dicht markiert werden, machen sie die Bemühungen des Trainings zunichte. Daher mussten die Patches manuell einzeln annotiert werden. Insgesamt wurden 7732 solcher Patches annotiert. Dazu wurde folgende Methode verwendet, die sich als äußerst effektiv und zuverlässig erwiesen hat:

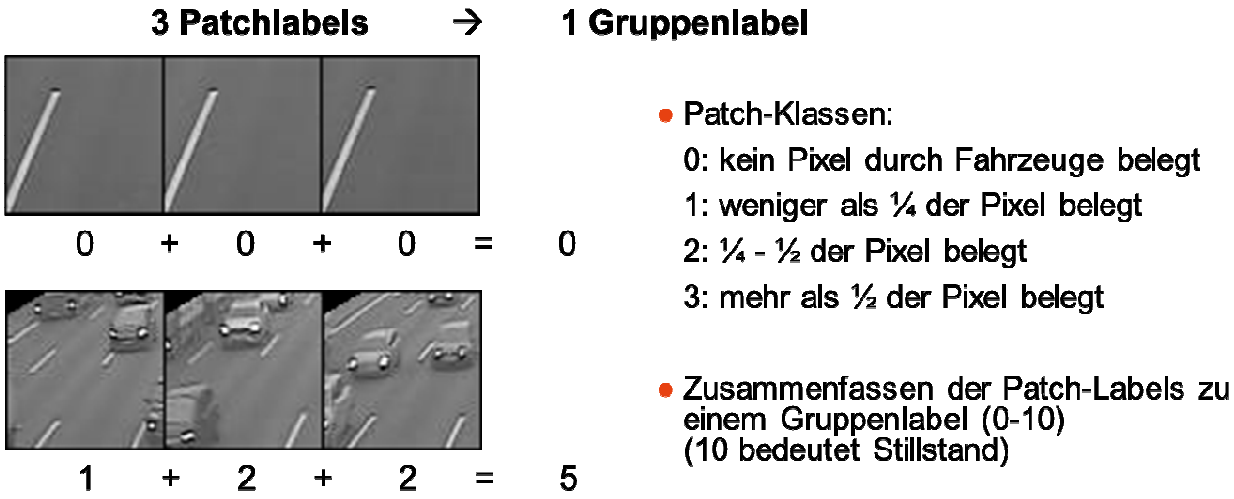

Abbildung 14: Annotierung der Trainingspatches

Jeder einzelne Patch wurde mit einem Wert von 0 bis 3 annotiert, wobei der Wert grob die Verkehrsdichte in dem Patch beschreibt. Der Gesamtwert für die drei Patches ergibt sich dann aus der Summe der Einzelbewertungen. Dieser konnte also zwischen 0 und 9 variieren. Zusätzlich wurde die Klasse 10 eingeführt um den speziellen Fall des Stillstands markieren zu können.

2.2.5 AP5 Evaluierung

Um das Kontinuum der Verkehrslagen optimal auf Klassen abzubilden sind neben den Klassen frei-dicht-gestaut noch Zwischenklassen eingeführt worden, siehe Kapitel 2.2.4. Dazu kam noch die Klasse Stillstand. Diese Klassen überlappen sich sehr stark. Um die Algorithmen besser beurteilen zu können, werden bei den Evaluationen Fehlerklassen verwendet. Für eine Evaluation werden die manuell ermittelten Klassen mit den automatisch geschätzten Klassen verglichen. Unter der Fehlerklasse 0 werden dabei alle Abweichungen als Fehler gewertet. Unter der Fehlerklasse 1 hingegen werden Abweichungen nicht gewertet, wenn die Klassen nur um 1 auseinander liegen, also direkt benachbart sind.
Der erste Algorithmus, der evaluiert worden ist, war der Median Algorithmus. Dazu wurde ein Datensatz verwendet, der über einen 24 Stunden Zeitraum an einem Werktag aufgezeichnet worden ist. Er stellt somit einen repräsentativen Datensatz dar, was Tageszeiten und Verkehrssituationen angeht. In Tabelle 1 sieht man einen Vergleich des Median Algorithmus mit festem Schwellwert für die Binarisierung und mit adaptiven Schwellwert nach Otsu. Mit adaptiven Schwellwert erreicht der Algorithmus eine Genauigkeit von 99,5% in der Fehlerklasse 1. Eine derart hohe Genauigkeit muss jedoch kritisch hinterfragt werden.

Algorithmus	Fehlerklasse 0	Fehlerklasse 1
Median	57,1%	85,0%
Median + Otsu	60,5%	99,5%

Tabelle 1: Evaluation Median Algorithmus mit repräsentativen Datensatz

Tatsächlich zeigte sich, dass der Median Algorithmus hervorragend arbeitet, solange der Verkehr frei oder frei-dicht ist. Ab dichtem Verkehr bricht die Genauigkeit jedoch stark ein. Die typische und bei weitem häufigste Verkehrslage ist jedoch frei bzw. frei-dicht. Daher stellt der repräsentative Datensatz den Algorithmus deutlich optimistischer dar als er tatsächlich ist.

Für die weiteren Evaluierungen wurde daher Datensätze zusammengestellt, in dem die dichteren Verkehrslagen in etwa das gleiche Gewicht hatten wie frei und frei-dicht.

Für die nächste Evaluierung wurde der CNN Algorithmus bewertet. Dieser liefert für jeden Patch eine Klasse zwischen 0 und 10. Auch für das Training erwies sich ein ausgeglichener Datensatz als essentiell. Nach zahlreichen Optimierungen und Verbesserungen ergab sich die Verwechslungsmatrix in Abbildung 15.

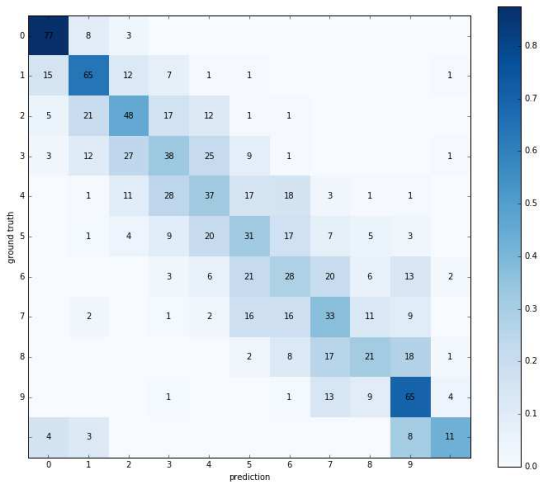

Abbildung 15: Evaluation der Patchklassen des CNN

Die Verwechslungsmatrix zeigt eine deutliche Hauptdiagonale mit den meisten Fehlern in den ersten Nebendiagonalen. Das war eine gute Basis für die weiteren Schritte.

Zwei Varianten für die Kombination der einzelnen Patchklassen zur kombinierten Sequenzklasse wurden evaluiert: Mittelwert und Neuronales Netz. In Abbildung 16 sieht man im Vergleich die Ergebnisse der drei Algorithmen Median, CNN mit Mittelwert und CNN mit neuronalem Netz.

Algorithmus	Fehlerklasse 0	Fehlerklasse 1
Median + Otsu	47,52%	81,79%
CNN + Mittelwert	41,10%	84,07%
CNN + NN	75,64%	92,90%

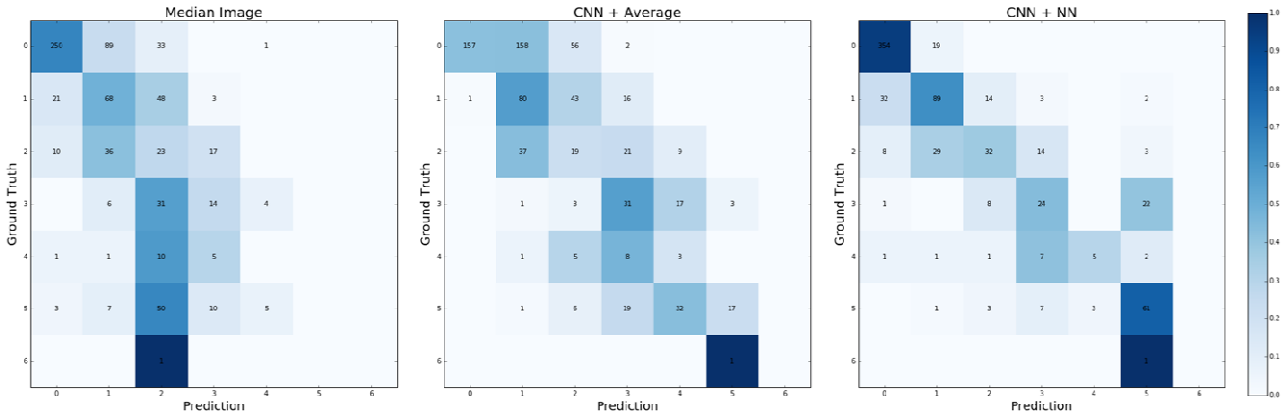

Abbildung 16: Ergebnisse der drei Ansätze

Das CNN mit Neuronalem Netz liefert die bei weiten besten Ergebnissen, während der Median Ansatz ab dichtem Verkehr sehr stark nachlässt.

Da der CNN Ansatz deutlich komplexer ist als der Median Ansatz, müssen auch die benötigten Rechenzeiten berücksichtigt werden.

Algorithmus	Dauer pro ROI	ROIs / Minute	Sequenzen / Minute*
Median Image	95 ms	630	315
CNN + Average	2,9 s	20	10
CNN + NN	2,9 s	20	10

Tabelle 2: Rechenzeiten der drei Ansätze

Tabelle 2 zeigt die Rechenzeiten der drei Ansätze. Der Zeitbedarf der Algorithmen wurde auf einem HP ProLiant Server mit Intel Xeon 3,1GHz, 4 Kernen und 32GB RAM gemessen. Dabei lief allerdings nur ein Data Analyzer. Das heißt, nur ein Kern wurde für die Berechnungen verwendet.

Wie erwartet war der Median Ansatz der bei weitem effizienteste. Tatsächlich könnten mit nur einem Kern bis zu 315 Kameras überwacht werden. Mit den CNNs sind es dagegen nur 10 Kameras je Kern. Das ermuntert zu folgender Kombination: Der Median Ansatz wird auf jeder Sequenz gerechnet. Ist die Verkehrslage frei oder frei-dicht, wird das Ergebnis übernommen, ansonsten werden die CNNs gerechnet.

2.2.6 Ausblick

Im Rahmen des Projekts wurde ein System entwickelt, welches in regelmäßigen Abständen Bilder von ausgewählten Webcams abfragt und Verkehrslageinformationen ableitet. Ziel war es, die prinzipielle Machbarkeit zu zeigen und die erreichbare Genauigkeit abzuschätzen.

Es zeigte sich, dass das System die Verkehrslage in den von den Kameras überblickten Straßenabschnitten mit hoher Zuverlässigkeit ermitteln kann. Allerdings sind diese Abschnitte im Vergleich zur Gesamtlänge der Straßen noch immer sehr klein. Die so gewonnenen Informationen sind eher lokal und eine Form der Glättung wird sicherlich noch anzuwenden sein. Es kam vor, dass die Kameras freien Verkehr meldeten obwohl die anderen Systeme Stau anzeigten. Auf den Bildern war dann tatsächlich kurzfristig freier Verkehr zu sehen. Innerhalb des Staus hatte es also Lücken gegeben, in denen die Fahrzeuge frei fahren konnten.

Um das System zu einem produktiven Einsatz weiterzuentwickeln sind noch einige weitere Schritte notwendig.

Das System wurde mit Daten von zahlreichen Kameras zu verschiedensten Tages- und Nachtzeiten sowie Witterungsbedingungen und Verkehrslagen trainiert und evaluiert. Dennoch gibt es einige Situationen, die noch nicht getestet werden konnten, da die Daten im Wesentlichen im Frühjahr und Sommer aufgezeichnet worden sind. Winterdaten und vor allem Schneelagen fehlen derzeit. Es ist damit zu rechnen, dass die CNNs mit einem erweiterten Trainingsdatensatz neu zu trainieren sind.

Derzeit werden alle eingehenden Daten in der Datenbank des Systems gespeichert. Dies führt allerdings dazu, dass die Datenbank mit der Zeit sehr groß und damit relativ langsam wird. Daher wird die Datenbank einmal pro Woche gelöscht. Eine intelligentere Garbage Collection wäre hier sinnvoll.

Das ganze Zusammenspiel mit den Systemen der ASFiNAG muss noch genauer definiert und implementiert werden. Wie und von wem sollen die ROIs definiert werden? Wie werden die Kameras neu konfiguriert, wenn Baustellen errichtet werden? Soll das System in das Monitoring der ASFiNAG eingebunden werden und wie soll dann die Schnittstelle aussehen? Werden für die Fusion mit den anderen Verkehrslagesystemen noch weitere Informationen benötigt? Wie soll das System skaliert werden?

Insbesondere das Thema der Skalierung ist in Hinblick auf den zukünftigen Ausbau genauer zu durchleuchten. Wenn die Zahl der Webcams von 660 auf über 1200 ansteigt, erhöht sich dementsprechend das Datenaufkommen und die erforderliche Rechenleistung. Die neuen Kameras werden aber auch eine höhere Auflösung (bis zu HD) aufweisen und sie werden tiefer geneigt sein, so dass sie weniger Himmel und mehr Fahrbahn aufnehmen. Dadurch werden die Bilder und die ROIs deutlich größer, was den Rechenaufwand weiter in die Höhe treibt.

Prinzipiell ist das höhere Datenaufkommen vor allem eine Frage der Bandbreite der Datenverbindung sowie der verfügbaren Rechenleistung. Die Performance des Medianansatzes ist dagegen von der Auflösung, mit der die Fahrzeuge dargestellt werden, relativ unabhängig. Die CNNs dürften hingegen sogar profitieren, wobei hierzu allerdings ein Training mit erweitertem Datensatz notwendig wäre.

Um mit der Rechenleistung auszukommen, gibt es zahlreiche Strategien, wie zum Beispiel:

- Leistungsfähigere oder mehr Server
- Die CNNs nur auf jedem zweiten Bild der Sequenz rechnen
- Die hochaufgelösten Bilder auf eine Standardauflösung herunterskalieren
- Eine feste Zahl an Patches in der ROI verteilen
- Einen Teil der Datenbank im RAM halten

Und es gibt noch viele weitere Möglichkeiten, mit das zu erwartende erhöhte Daten-aufkommen zu verarbeiten.

2.3 Änderungen im weiteren Projektverlauf

Da das Projekt aufgrund vertraglicher Schwierigkeiten nicht zum 1.7.2015 starten konnte, wurde das Projektende auf den 31.7.2016 verschoben.

3 PROJEKTTEAM UND KOOPERATION

Das Projekt wurde im Wesentlichen vom Auftragnehmer, EFKON AG durchgeführt unter Mitwirkung der ASFiNAG. Das Kickoff Meeting fand am 5.8.2015 statt. Bis Ende 2015 wurde

über den Projektfortschritt in zweimonatigen Zwischenberichten berichtet. In 2016 wurden monatliche Statusbesprechungen durchgeführt, gegen Ende des Projekts auch häufiger. Diese fanden zumeist in Person, teilweise auch telefonisch bzw. per Skype statt.

4 WIRTSCHAFTLICHE UND WISSENSCHAFTLICHE VERWERTUNG

Die Ergebnisse wurden am 9. August 2016 der ASFiNAG präsentiert. Dabei wurde auch erläutert inwiefern die aus den Kameras generierten Informationen mit den Informationen aus den anderen Systemen korrelieren bzw. komplementär sind. Eine Strategie, wie diese Informationen am besten fusioniert werden können, war nicht Teil des Projekts und ist noch zu erarbeiten. ASFiNAG wird in Folge entsprechende Strategien untersuchen und den Business Value des Systems abschätzen. Hierzu wird das System noch bis Ende September in Betrieb gehalten.