



Department of Computer Engineering

Classification of Large Biological Image Sets within the Browser

Bachelor Thesis

Christoph Friedrich

Summer Term 2018

First Advisor: Prof. Dr. Jürgen Koch
Second Advisor: David Dao ETH Zürich

Declaration

Hereby I certify that I did this work by my own. Literally or analogously used intellectual property was declared as such.

Esslingen, May 14, 2018

Place, Date

Christoph Friedrich

Contents

1. Motivation	1
2. Grundlagen	2
2.1. Verwendete Hard- und Software	2
2.1.1. Hardware	2
2.1.2. Software	3
3. Systementwurf	4
3.1. Abtastung	4
3.1.1. Eleminiere Artefakte	5
3.2. Wahrscheinlichkeitsverteilung	5
3.2.1. Filter ROI	6
3.2.2. Filter Spurbreite	6
3.2.3. Filter Streifenbreite	6
3.2.4. Filter Tiefpass	6
3.2.5. Trafo	6
3.3. Szenarien	6
3.3.1. Ausbleibende Spur	6
3.3.2. Zebrastreifen	6
3.3.3. Kreuzung	6
3.3.4. Weitere denkbare Szenarien	6
3.4. Ausblick	6
3.4.1. Vortasten	6
3.4.2. Kalman Filter	6
4. Systemanalyse	7
4.1. Kontextdiagramm	7
4.2. Anwendungsfalldiagramm	7

4.3. Kurzbeschreibungen zu jeden Anwendungsfall	8
5. Systementwurf	9
5.1. Klassendiagramm	9
5.2. Logisches Datenmodell	9
5.3. Physikalisches Datenmodell	9
5.4. Entwurf der grafischen Oberfläche	9
6. Zusammenfassung	10
Bibliography	11
List of Figures	12
List of Tables	13
Listings	14
A. Anhang zum Systementwurf	15
A.1. Diagramme	15
A.2. Tabellen	15
A.3. Quellcodelistings	15

1. Motivation

These days a lot of data is generated to improve the way we use our cars or to unlock our phones, autonomous driving and face recognition are only two examples where we need a lot of data to improve algorithms and with that the devices we use that algorithms for. But the maybe most interesting area where a lot of data is collected is health care. Modern technologies give us possibilities we never had before, artificial intelligence makes it possible to determine if one has a malignant melanoma or just a harmless mole with your cellphone. However there is still a lot of high potential data unused, one reason for that is most people don't know how to use the data they have. It seems to be a privilege to Computer Scientists and Computer Enthusiasts to take advantage of that data and the potential it holds. Imagine biologists and doctors going through Hundreds and Thousands of samples, giving them a tool to automatize there daily work flow would be a huge ease. Goal of this bachelor thesis is to give normal users, users without knowledge of how to install a Python environment or dealing with databases the possibility to use all advantages of machine learning. By that this projects somehow follows the modest aim to democratize artificial intelligence.

2. Grundlagen

In diesem Kapitel werden die wichtigen Grundlagen für diese Aufgabenstellung erläutert.

2.1. Verwendete Hard- und Software

2.1.1. Hardware

Alle hier vorgestellten Algorithmen laufen auf einem Raspberry Pi 3 Modell B mit einer Quad Core 1.2GHz Broadcom BCM2837 64bit CPU und 1GB RAM [3].

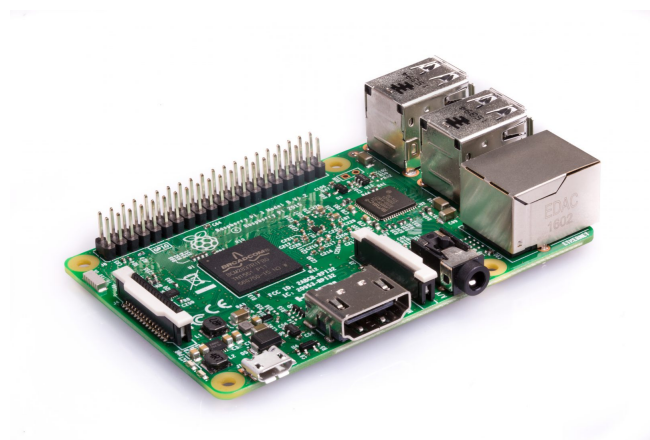


Figure 2.1.: Raspeberry Pi 3 Modell B, Quelle:[4]

Weiter wird eine Kamera von Raspberry Pi Camera Module v1 mit einer Auflösung von 640 x 480p bei 60 fps verwendet [2].

2.1.2. Software

Zur Programmierung der Software wird MATLAB® (2017b) mit Simulink® und dessen Embedded Coder™, sowie die Image Processing Toolbox™ verwendet.



Figure 2.2.: MATLAB Logo

3. Systementwurf

3.1. Abtastung

Idee ist es, das aufgenommene Bild in vier verschiedene Regionen einzuteilen. (Siehe Bild 3.1)

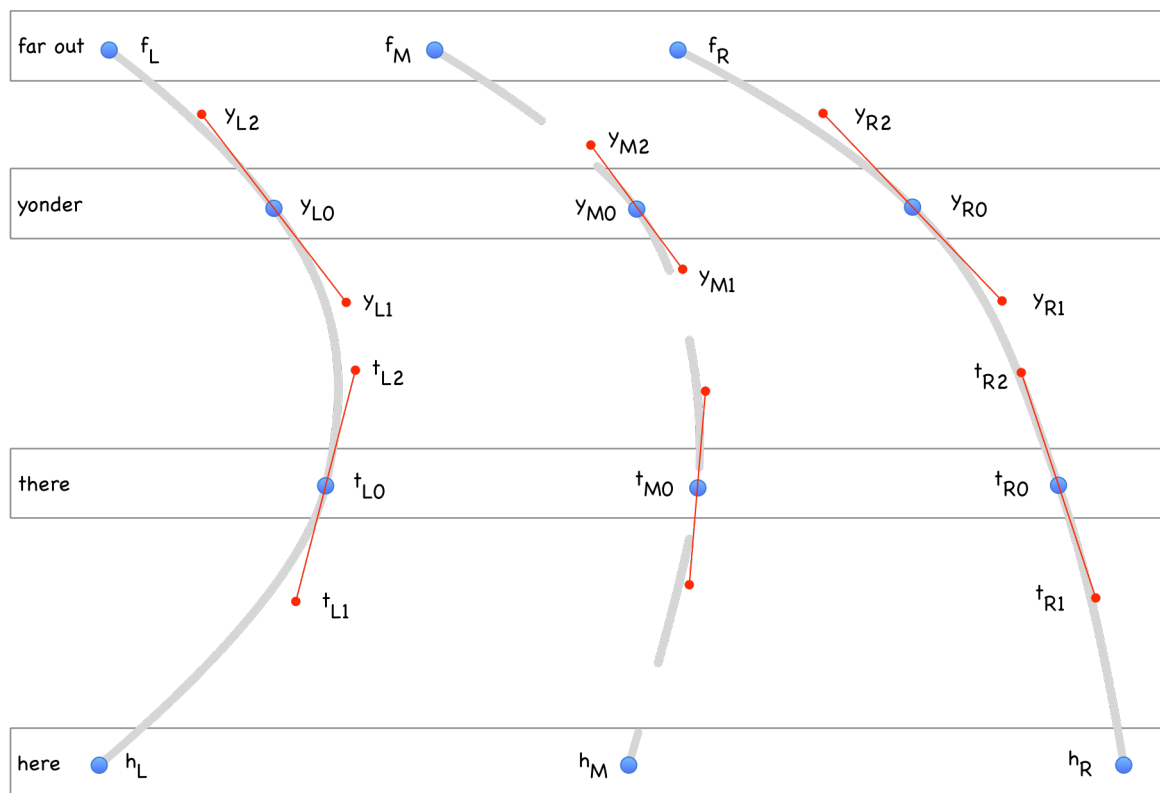


Figure 3.1.: Verschiedene Regionen

Für jede dieser Regionen wird jeweils eine Zeile abgetastet. Als Ergebnis erhält man ein Histogramm (Siehe Bild 3.1). Man kann erkennen, dass die Spurmarkierungen durch Peaks mit einer bestimmten Breite und einem bestimmten Abstand (in Grün) zueinander erkennbar sind. In Rot zu erkennen sind Artefakte, welche sich durch eine kleine Breite(1-2px) auszeichnen. Sie entstehen zum Beispiel durch Reflexionen auf der Fahrbahn.

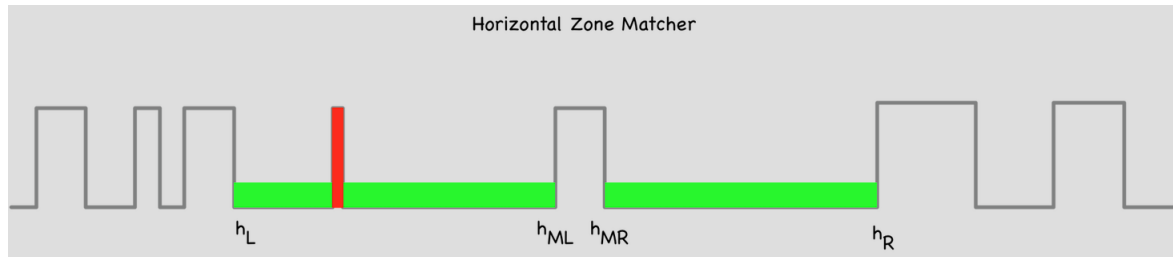


Figure 3.2.: Histogramm

3.1.1. Eleminiere Artefakte

Um Artefakte und Objekte ohne Kante zu eliminieren werden die abgetasteten Regionen mit der Canny Kantenerkennung gefiltert. Übrig bleiben Punkte, die zu einer Kante gehören und deutlich weniger Artefakte.

3.2. Wahrscheinlichkeitsverteilung

Für jeden gefundenen Punkt in der Region soll bestimmt werden mit welcher Wahrscheinlichkeit dieser zu einer Spurmarkierung der rechten Fahrspur gehört. Verschiedene Filter werden dafür eingesetzt, jeder dieser Filter addiert eine Wahrscheinlichkeit zu jedem der gefundenen Punkte. Initial erhalten alle Punkte die Wahrscheinlichkeit null.

3.2.1. Filter ROI

3.2.2. Filter Spurbreite

3.2.3. Filter Streifenbreite

3.2.4. Filter Tiefpass

3.2.5. Trafo

3.3. Szenarien

3.3.1. Ausbleibende Spur

Verhalten

3.3.2. Zebrastreifen

Zebrastreifen Erkennung

Verhalten

3.3.3. Kreuzung

Stopplinien Erkennung

Verhalten

3.3.4. Weitere denkbare Szenarien

3.4. Ausblick

3.4.1. Vortasten

3.4.2. Kalman Filter

4. Systemanalyse

Abbildung 4.1 zeigt den prinzipiell Ablauf der Systemanalyse.

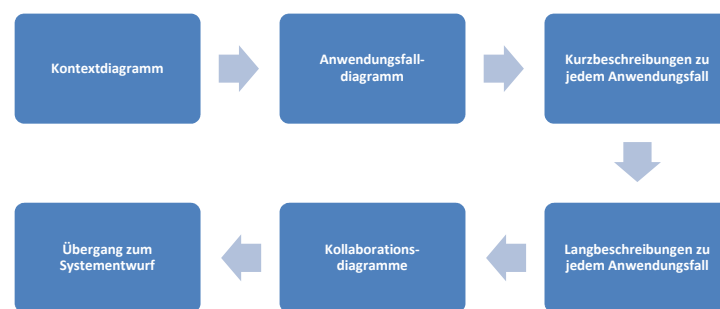


Figure 4.1.: Ablauf der Systemanalyse

4.1. Kontextdiagramm

...

4.2. Anwendungsfalldiagramm

...

4.3. Kurzbeschreibungen zu jedem Anwendungsfall

...

5. Systementwurf

5.1. Klassendiagramm

5.2. Logisches Datenmodell

5.3. Physikalisches Datenmodell

5.4. Entwurf der grafischen Oberfläche

6. Zusammenfassung

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language. Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Bibliography

- [1] Ekbert Hering, Rolf Martin und Martin Stohrer. *Physik für Ingenieure*. Springer Verlag, 2004.
- [2] RASPBERRY FOUNDATION. Cam module v2, 2016. Quelle: <https://www.raspberrypi.org/products/camera-module-v2/>, zugegriffen am 15.12.2017.
- [3] RASPBERRY FOUNDATION. Raspberry 3 modell b, 2016. Quelle: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>, zugegriffen am 15.12.2017.
- [4] RASPBERRY FOUNDATION. Raspberry 3 modell b, 2016. Quelle: <https://www.raspberrypi.org/app/uploads/2017/05/Raspberry-Pi-3-462x322.jpg>, zugegriffen am 15.12.2017.
- [5] Joachim Goll, Cornelia Heinisch und Frank Müller. *Java als erste Programmiersprache*. Teubner Verlag, 2005.
- [6] Lothar Papula. *Mathematische Formelsammlung für Wissenschaftler und Ingenieure*. Vieweg Verlag, 2003.
- [7] Manfred Dausmann, Ulrich Bröckl und Joachim Goll. *C als erste Programmiersprache*. Teubner Verlag, 2005.

List of Figures

2.1. Raspeberry Pi 3 Modell B, Quelle:[4]	2
2.2. MATLAB Logo	3
3.1. Verschiedene Regionen	4
3.2. Histogramm	5
4.1. Ablauf der Systemanalyse	7

List of Tables

Listings

A. Anhang zum Systementwurf

Allgemeine Beschreibung des Anhangs

A.1. Diagramme

Hier werden Diagramme platziert, die in den Textkapitel zuviel Platz beanspruchen.

A.2. Tabellen

Hier werden Tabellen platziert, die in den Textkapitel zuviel Platz beanspruchen.

A.3. Quellcodelistings

Hier werden Tabllen platziert, die in den Textkapitel zuviel Platz beanspruchen.