

# VOR-ORT AUFNAHME UND MATLAB BASIERTE AUSWERTUNG VON EINGANGSDATEN FÜR PERSONENSIMULATOREN

Praktikant: Schwiebacher Johannes

Lehrstuhl für Computergestützte  
Modellierung und Simulation

Betreuer: Kielar Peter

Zeitraum der Tätigkeit: 15.12.2014-  
15.04.2015

Datum der Abgabe

15.04.2015

## Inhalt

|                              |   |
|------------------------------|---|
| 1. Einleitung.....           | 3 |
| 2. Perspektivenrechnung..... | 3 |
| 3. Erstellung einer GUI..... | 4 |
| 4. Fußgängersimulation.....  | 5 |
| 4.1 Datenerhebung.....       | 5 |
| 4.2 Datenauswertung.....     | 6 |
| 5. Fazit .....               | 7 |

## 1. Einleitung

Das Forschungspraktikum wurde im Feld der Fußgängersimulation am Lehrstuhl für Computergestützte Modellierung und Simulation durchgeführt. Bei der Fußgängersimulation werden mittels Computermodellen Bewegungsverhalten von Fußgängern nachempfunden. Hierzu ist es notwendig Daten zu erheben um die Korrektheit der erstellten Modelle zu überprüfen. Im Kontext der Datenerhebung führte ich einige Aufgaben durch, welche nachfolgend erläutert werden.

## 2. Perspektivenrechnung

Die erste Aufgabenstellung bestand in der Rückberechnung eines perspektivisch verzerrten Bilds in dessen Ausgangszustand.

In diesem Kontext wurde ein Video verwendet, das zur Analyse von Fußgängerbewegungen erstellt wurde. Abbildung 1 zeigt eine Aufnahme dieses Videos. Die Aufgabe bestand darin, das durch Verkehrsleitkegel abgesteckte Feld aus der verzerrten Perspektive, in dessen Originalzustand eines Quadrats zurückzurechnen.

Zunächst zog ich zur Bearbeitung den trigonometrischen Ansatz heran. Nachdem jedoch Daten, wie beispielsweise die Höhe der Kamer oder der Abstand der Kamera zu den Kegeln fehlten, gab ich diese Idee auf.



Abbildung 1: Aufnahme Fußgängerbewegung

So widmete ich zu Beginn des Praktikums einige Zeit dem Finden eines geeigneten Ansatz zur Lösung der Problemstellung.

Um das Viereck so auszurichten, dass an den Ecken rechte Winkel entstehen, verwendete ich Translations-, Rotations- und Verzerrungsmatrizen. Doch auch dieser Ansatz zeigte sich nicht als erfolgreich.

Schließlich versuchte ich eine Koordinatentransformation. Obwohl die Koordinatentransformation nicht möglich war, konnten so verschiedene Ansätze ausgeschlossen werden.

### 3. Erstellung einer GUI

Die zweite große Aufgabe bestand in der Abänderung einer Benutzeroberfläche. Hierzu erhielt ich vom Lehrstuhl eine bereits vorgefertigte GUI, in welcher es möglich war Videos einzulesen, sowie von Frame zu Frame zu wechseln. Ich sollte diese GUI anschließend um folgende Funktionen erweitern:

Es sollte möglich sein, in dem eingelesenen Video die Verkehrsleitkegel zu markieren. Außerdem sollte eine Funktion erstellt werden, welche sowohl die Abstände der markierten Punkte, als auch die Winkel zwischen den Punkten automatisch berechnet.

Nachdem das Programm mittels eines Radiobuttons in den Kalibrierungsmodus gestellt wurde, konnten per Klick auf das Video die Kegelköpfe markiert werden (siehe Abbildung 2).



Abbildung 2: Markierte Verkehrsleitkegel durch den Toggle-Button

Die Berechnungen wurden über Vektoren- und trigonometrische Rechnungen- verwirklicht. Die markierten Punkte wurden hierfür in das System eingelesen und in einer Matrix abgespeichert. Das Programm erstellt aus den gespeicherten Punkten Vektoren. Diese Vektoren können dann sowohl zur Berechnung der Winkel, als auch der Abstände zwischen den Punkten herangezogen werden.

Dies war der erste Schritt zur Lösung der Verzerrungsumrechnung. Mittels dieser Funktion können Berechnungen im Bild erfolgen, zum Beispiel wie viele Bildpixel einem Meter entsprechen.

## 4. Fußgängersimulation

### 4.1 Datenerhebung

Die dritte Aufgabenstellung widmete sich einem anderen Themengebiet.

Zur Simulation des Fußgängerstroms der Münchner U-Bahn Haltestelle „Garching Forschungszentrum“, zählte ich mittels einer vom Lehrstuhl geschriebenen App dort die Fahrgäste .



Abbildung 3: Luftbild Garching Forschungszentrum

Die App funktioniert durch das simple Klicken auf den Touchscreen des Smartphones.

Die App speichert die Klicks zeitgestempelt. Im Bild sieht man die Oberfläche der App, welche an sich selbsterklärend ist:

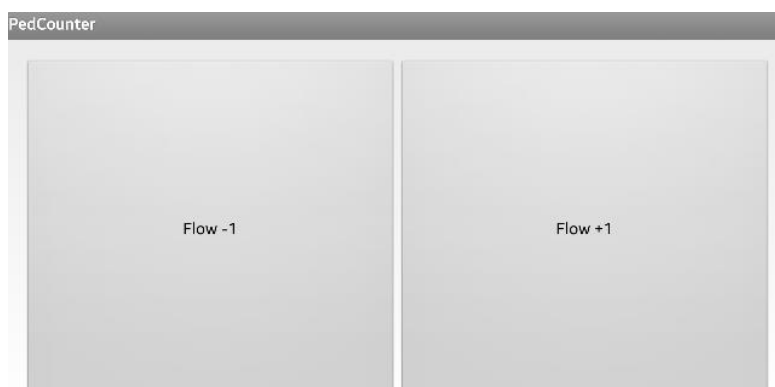


Abbildung 4 Oberfläche App

## 4.2 Datenauswertung

Nun galt es die erhobenen Daten mittels einer Matlab Funktion auszuwerten und in einem Histogramm darzustellen.

Nach der Datenerhebung erhielt ich Datensätze folgender Form:

„1;13999#2;14026#3;14035#4;14036#5;14037#6;14038#7;14039#8;14040#9;14041“

Bei diesem Datensatz steht die erste Zahl für die Nummer des Klicks auf dem Touchscreen. Die 2. Zahl nach dem Semicolon für die Zeit. Die Zeit wird hier in einer speziellen 5-stelligen Kodierung (Stunden\*3600+Minuten\*60+Sekunden) ausgeführt, relevant sind dabei vor allem die letzten zwei Ziffern. Sie stellen die Sekunden des Zeitpunktes des Klicks dar. Diese Kodierung funktioniert nicht, wenn während der Messung der Tag wechselt, dies war bei den erhobenen Daten jedoch nicht der Fall.

Die Matlabfunktion ist wie folgt strukturiert:

Zunächst wurde mit nur einem Datensatz begonnen, um das Programmieren zu erleichtern. Ich las den Datensatz ein, und speicherte ihn als Matrix erneut ab.

Abbildung 5 zeigt den nächsten Vorgang. Die Matrix veränderte ich so, dass in der linken Spalte die Zeit und in der Rechten die Anzahl der Personen in diesem Zeitintervall dargestellt werden.

|    |   |
|----|---|
| 0  | 1 |
| 5  | 1 |
| 9  | 1 |
| 10 | 1 |
| 12 | 2 |
| 13 | 2 |
| 15 | 2 |
| 17 | 1 |
| 18 | 1 |
| 20 | 2 |
| 21 | 2 |

Auch diese Matrix wurde als CSV Datei abgespeichert, um das spätere Begutachten einzelner Datensätze zu ermöglichen.

Abbildung 5 Datensatz Matrix

Abbildung 6 zeigt das Resultat aller erhobenen Datensätze. Die rote Kurve beschreibt die Normalverteilung. Der Balken bei 0 entsteht durch den ersten Klick beim Einfahren der U-Bahn.

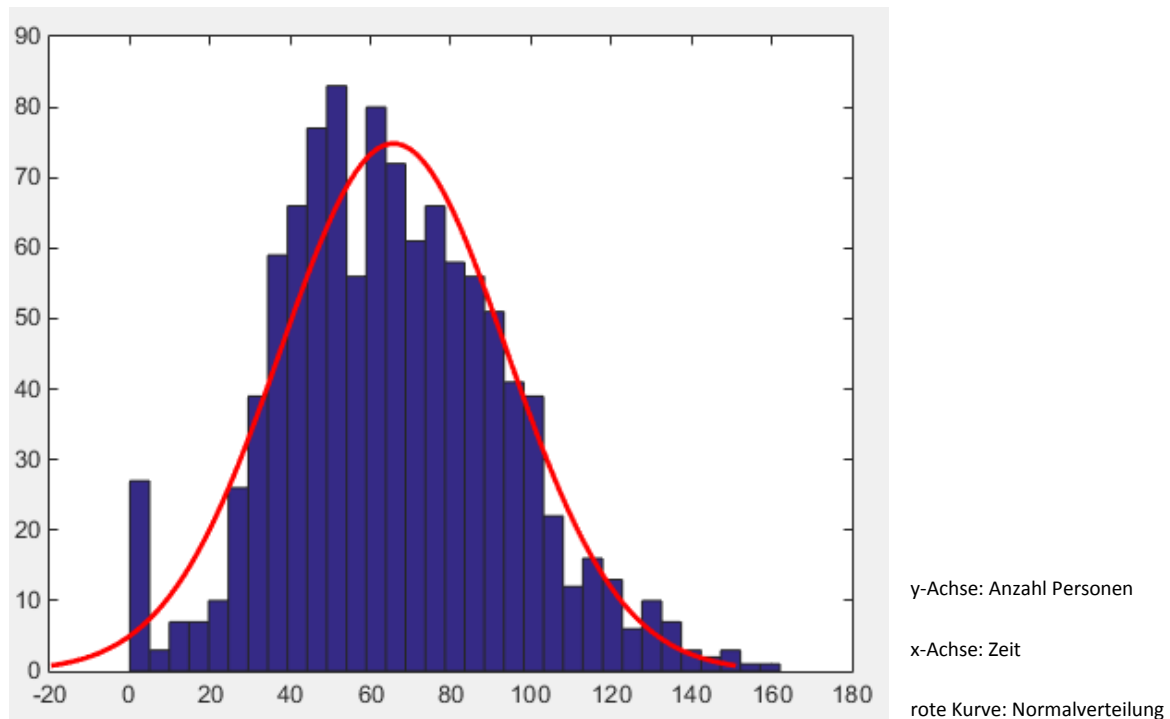


Abbildung 6 Histogramm

## 5. Fazit

Das Praktikum war sehr lehrreich und interessant. So konnte ich im Bereich der Fußgängersimulation sogar mein eigenes kleines Forschungsergebnis erzielen, und einen Beitrag zur Arbeit meines Betreuers liefern.

Ich konnte außerdem mein Wissen zum Programm „Matlab“ erweitern und verbessern, sowie generell einen Einblick in die Forschungstätigkeiten der TU München erhalten, und den Lehrstuhl mit Validierungsdaten unterstützen.