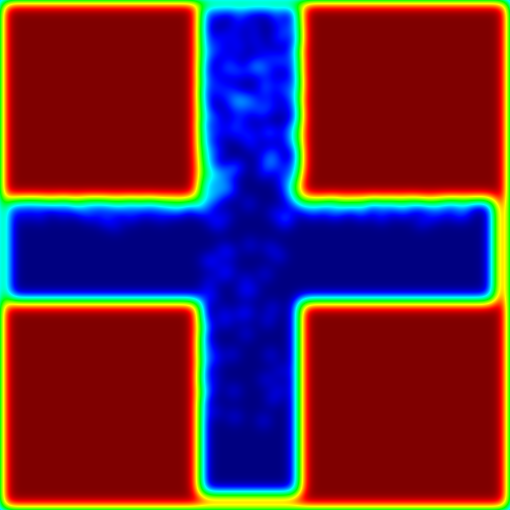
Technischer Bericht aus dem Modellierungsseminar

„Can we learn where people go?“



Gruppe: Datenerzeugung und Datenaufbereitung

Sprint: 2.Sprint

Mitglieder: Julian Bauer

Rebecca Elisabeth Brydon

Lisa-Marie Grundmann

Hubert Hager

Do Nguyen

Amir Schnell

Anita Steinberger

Inhaltsverzeichnis

[1. Aufgabenstellung und Ziele des ersten Sprints 1](#_Toc500953595)

[2. Sprintziele des 2. Sprints 4](#_Toc500953596)

[2.1 Das Modellieren realistischer, variantenreicher Szenarien 5](#_Toc500953597)

[2.2 Die Filter in der Datenvorverarbeitung 10](#_Toc500953598)

[2.3 Die Verteilung der Personen in VADERE 16](#_Toc500953599)

[2.4 Die Datenformate und die Datengenerierung 18](#_Toc500953600)

[2.5 Das Testen unserer Software 22](#_Toc500953601)

[2.6 Optimierung des bisherigen Codes 25](#_Toc500953602)

[2.7 Weitere Ziele des 2. Sprints 26](#_Toc500953603)

[3. Implementierungen 27](#_Toc500953604)

[4. Zusammenfassung und Ausblick in den 3. Sprint 28](#_Toc500953605)

Tabellenverzeichnis

[Datenformat der Trajektorien 2](#_Toc500953606)

[Tabelle mit Anzahl an Singulärwerten und entsprechendem MSE 14](#_Toc500953607)

[Fraunhofer Datenformat 20](#_Toc500953608)

Abbildungsverzeichnis

[Graphische Darstellung der Personendichte und eines Szenarios 2](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953610)

[Lageplan der Canstetter Wasen 5](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953611)

[Lageplan der Canstetter Wasen samt eingezeichneten interessanten Bereich 5](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953612)

[Intesessanter Bereich der Wasen im Detail 6](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953613)

[Umgesetzte Szenarien der Wasen 6](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953614)

[Detaillierte Angabe der Schwierigkeiten des Wasen-Szenarios 7](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953615)

[Weiterer interressanter Wasen-Ausschnitt 7](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953616)

[Umgesetztes Wasen-Szenario 8](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953617)

[Szenario für die Gruppen des Maschinellen Lernens 8](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953618)

[Szenario für die Gruppen des Maschinellen Lernens 9](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953619)

[Szenario für die Gruppen des Maschinellen Lernens 9](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953619)

[Dichtebild einer Beispielmatrix 12](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953620)

Singulärwerte der Beispielmatrix…………..........…………………………………………………….12

[Reduzierte Matrix mit 99% der Originalinformation………………………………………………….13](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953621)

[Reduzierte Matrizen mit 95%, 90%, 75% und 50%n der Originalinformation 13](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953622)

[Reduzierte Matrix mit vorheriger Zentrierung und 99% der Originalinformation 14](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953623)

[Reduzierte Matrix nach vorheriger Subrtaktion des Durchschnitts und 99% der Originalinformation 15](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953624)

[Ausschnitt aus einer VADERE Simulation 16](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953622)

[Code-Ausschnitt zur Berechnung der zufälligen Verteilung 16](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953623)

[Code-Ausschnitt zur Berechnung der Permutationen 17](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953622)

[Szenario-Ausschnitt samt Kameraposition…………………………………………………………….18](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953623)

[Trajektorien vor der Formatierung 20](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953622)

[Trajektorien nach der Formatierung 20](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953623)

[Code-Ausschnitt zur Berechnung des Geschwindigkeitsvektors……………………………………………………….2](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953623)1

[Vergleich der Dichtebilder und der Postvisualisierung in VADERE 2](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953622)2

[Vergleich der Dichtebilder und der Postvisualisierung in VADERE 2](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953622)2

[Vergleich der Dichtebilder und der Postvisualisierung in VADERE ………………………………………………..2](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953623)3

[Vergleich Dichtebild vor und nach der Bearbeitung durch den Binärisierungsfilter 23](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953622)

[Histogramm der zufällig berechneten Werte bei der Verteilung 24](file:///C:\Users\Lisa-Marie\Desktop\letztes%20Semester\Modellbildungsseminar\Technischer%20Bericht_Datengruppe.docx#_Toc500953622)

# Aufgabenstellung und Ziele des ersten Sprints

In dem Projektstudium „Modellierungsseminar“ wird in diesem Semester versucht, die Frage „can we learn where people go“ (zu Deutsch können wir herausfinden, wohin die Menschen gehen) zu beantworten. Dies geschieht mit Hilfe von 3 Gruppen, der Datenerzeugungs- und Datenanalysegruppe, sowie zwei Gruppen des Maschinellen Lernens, der Neuronale Netze Gruppe und der Random Forrest Gruppe.

Mit Hilfe des Simulationsframeworks VADERE der Hochschule München sollen in diesem Rahmen Daten erzeugt werden, mit deren Hilfe die Frage beantwortet werden kann, ob wir Bewegungsmuster lernen können, wohin Menschen gehen. Die Ergebnisse sollen weiterhin so dargestellt werden, dass mögliche neue Studenten auf den Studenteninformationstag Spaß an dieser Sache finden, außerdem soll ein Poster für die PED 2018 (**P**edestrian and **E**vacuation **D**ynamics) erstellt werden.

In dem ersten Sprint ging es vorrangig um die Installation und Einarbeitung in die für die Umsetzung der Ziele des ersten Sprints benötigte Software bzw. Programmiersprachen. Für unsere Gruppe der Datenerzeugung und Datenanalyse war dies vor allem die Einarbeitung in VADERE und Python. Es sollten mittels VADERE einige Szenarien erstellt werden, in denen die Menschen zwischen verschiedenen Zielen wählen und die Daten, die dabei mit Hilfe von VADERE erzeugt worden sind, sollten so aufbereitet werden, dass sie den Gruppen des Maschinellen Lernens zum Lernen übergeben werden konnten. Im ersten Sprint wurde sich zunächst mit den Gruppen des Maschinellen Lernens auf erste Datenformate geeinigt. Zum einen war dies ein Datenformat, das aus den Trajektorien bestehen sollte und das die Verteilung der Personen prozentual angab.

Da das Ziel der Gruppen des Maschinellen Lernens darin besteht, den Weg, den eine Person einschlagen wird, vorauszusehen, sollte natürlich nicht der gesamte Weg aufgezeichnet werden, den eine Person auf ihrem Weg, von der Quelle zum Ziel hin bestritt, sondern es sollten die Koordinaten der Personen nur dann gespeichert werden, wenn sie sich in einem bestimmten „Kameraausschnitt“ (oder Kamerabild) befanden. Dieser Ausschnitt wurde so in die jeweiligen Szenarien eingebaut, dass er vor Kreuzungen bzw. Abzweigungen lag. Die Aufgabe der Gruppen des Maschinenlernalgorithmen bestand nun darin, durch die Koordinaten der Personen innerhalb des Ausschnitts und mit Hilfe der Verteilung den Weg, den eine Person nehmen wird, vorherzusagen.

Der Aufbau dieses Formats war wie folgt:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Timestep | Person 1 | Person 1 | … | Person m | Person m | Verteilung |
| 1 | x-Koordinate | y- Koordinate |  | x- Koordinate | y- Koordinate | (t1%, …, tk%) |
| … | x-Koordinate | y- Koordinate |  | x-Koordinate | y- Koordinate | (t1%, …, tk%) |
| n | x-Koordinate | y- Koordinate |  | x-Koordinate | y- Koordinate | (t1%, …, tk%) |

Tabelle 1

Die Tabelle soll das Datenformat aus den Trajektorien des ersten Sprints veranschaulichen. In jedem Schritt (Timestep) wird die x- und y-Koordinate der Personen, die in diesem Szenario laufen, angegeben. Sollte eine Person in einem Schritt bereits das Ziel erreicht haben, noch gar nicht losgelaufen sein, oder außerhalb des „Kamerabildes“ sein, so werden für die x- und y-Koordinate der Person in diesem Schritt die Werte (-1,-1) gesetzt. In der Spalte „Verteilung“ wurde ein Vektor angegeben, in dem prozentual gespeichert wurde, wie viele Personen sich auf welches Ziel verteilen. So gibt beispielsweise der Wert t1% in Prozent an, wie viele Menschen in diesem Schritt zu dem Ziel 1 gelaufen sind. Diese Matrix wurde am Ende den anderen beiden Gruppen zum Lernen übergeben.

Des Weiteren wurde sich auf ein Datenformat geeinigt, welches die Dichte aus dem Personenstrom berechnet. Dieses weitere und insbesondere andere Datenformat wurde festgelegt, um herauszufinden, mit welchem der Datenformate (berechnet aus den Trajektorien, oder berechnet aus der Dichte) die Gruppen des Maschinellen Lernens besser lernen können. Die Messfläche wird dafür in quadratische Raster diskretisiert. Die Dichte jeder Person wird mit Hilfe der Gauß Dichtefunktion für jedes Rasterfeld angenähert, wobei die Dichtefunktion der Bachelorarbeit von Herrn Zönnchen entnommen wurde. Die symmetrische Glocke wird erst abgeschnitten, wenn die Dichtewerte annähernd null erreichen. Die Werte der Dichte werden für alle Personen auf dem Rasterfeld pro Zeitschritt aufaddiert. Die Größe der Raster wird über den *resolution* Parameter bestimmt, dieser wurde nach Besprechung mit den Maschine Learning Gruppen auf r = 0.5 gesetzt.

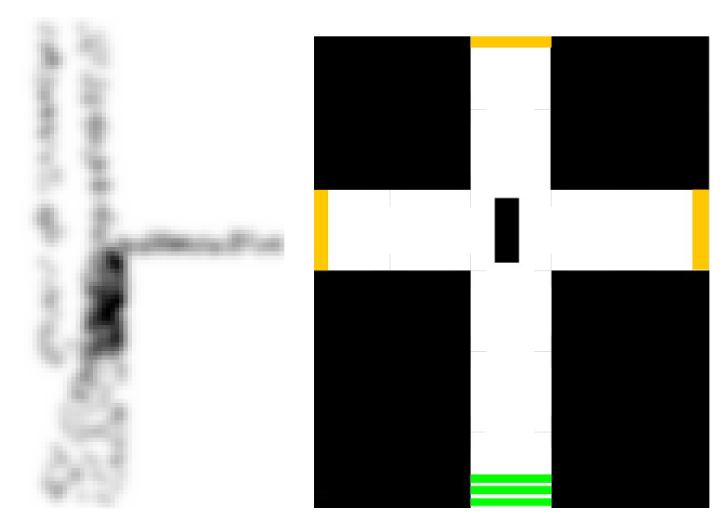


Abbildung 1

Graphical representation of the density on the left hand side and on the right hand side one of our scenarios.

Da die Gruppen des Maschinellen Lernens eine sehr große Mengen an Daten mit unterschiedlichen Verteilungen benötigten, ergab sich während des ersten Sprints noch das Ziel, diese Simulation in VADERE zu automatisieren, denn das per Hand in VADERE umzusetzen, erforderte sehr viel Zeit. Die Idee bei hierbei war, ein Programm zu schreiben, was von außerhalb VADERE startet und die entsprechenden Werte, die für die Simulation benötigt werden mehrmals neu setzt. Es ergab sich bei der Umsetzung dieses Ziels zunächst das Problem, dass die entsprechenden Parameter von außerhalb VADERES zur Laufzeit nicht verändert werden konnten. Dank der Hilfe von Herrn Zönnchen, konnte dieses Problem jedoch relativ schnell behoben werden, sodass am Ende des ersten Sprints die Automatisierung der Simulationen möglich war.

Ein weiteres Ziel des ersten Sprints war die erste Literaturrecherche über die Filter in der Datenvorverarbeitung, mit dem Ziel, einen ersten Eindruck dafür zu bekommen, welche Filter nützlich sein könnten, um mit dessen Hilfe unsere erzeugten Daten so aufzubereiten, sodass das Lernen mittels der Neuronalen Netze und des Random-Forrest Algorithmus besser funktioniert. Da sich über dieses Thema im ersten Sprint zunächst nur ein grober Überblick verschafft wurde, wurde die weitere Recherche und auch die Implementierung erster Filter als ein Ziel des zweiten Sprints angenommen.

Um eine gute Kommunikation innerhalb der Gruppe und das gemeinsame Arbeiten am Code zu gewährleisten sowie damit jedes Teammitglied zu jeder Zeit wusste, was er oder sie zu tun hat, hat sich unsere Gruppe ein eigenes Repository und ein eigenes Trello Projekt erstellt. Mittels des Repositories war das gemeinsame Arbeiten am Code zu jeder Zeit gewährleistet und mittels des Trello Projektes, in dem die Kanban-Wand unseres Teams immer auf dem aktuellen Stand gehalten wurde, war es für jedes Teammitglied immer einsehbar, was die Aufgaben eines Jeden während des Sprints waren. Des Weiteren wurde durch unsere regelmäßigen Treffen außerhalb der Vorlesung die Hilfe aller Teammitglieder bei Problemen des Einzelnen sichergestellt.

Das letzte Ziel des ersten Sprints war das Testen der von uns programmierten Software um die Korrektheit der von uns gelieferten Daten zu jedem Zeitpunkt gewährleisten zu können.

# Sprintziele des 2. Sprints

Natürlich gab es für den 2. Sprint neue Ziele, die auf den Zielen des ersten Sprints basierten und uns der Beantwortung der Frage „can we learn where people go?“ näherbringen sollten. Für unsere Gruppen waren dies zum einen die folgenden von Frau Prof. Dr. Köster vorgegebenen Ziele:

* Erstellen dieses technischen Berichts
* Modellieren und simulieren neuer realistischer, variantenreicher Szenarien
* Erstellen des Datenformats entsprechend dem vom Fraunhofer Institut vorgeschriebenen Formates
* Aufbereitung der Daten, dass diese denen von Kameras gewonnenen Daten stark ähneln
* Erstellen einer Demo für die zweite Sprintabnahme.

Zum anderen gab es auch von unserer Gruppe selbstgewählte Ziele, die in diesen Sprint umgesetzt werden sollten:

* Weitere Literaturrecherche über die Filter der Datenvorverarbeitung
* Die Implementierung einiger ausgewählter Filter
* Erstellen einer Dokumentation der bis dato eigens von uns implementierten Software
* Die Verteilungen der Personen, die bei der Automatisierung der Simulation in VADERE möglich sind, sollten durch Permutationen realisiert werden
* Weiteres testen unserer Software
* Die Datengenerierung
* Aktualisierung unseres Trello Projektes, unseres Repositories und die Organisation der Treffen.

Nachfolgend sollen die einzelnen Ziele des 2. Sprints näher erläutert werden. Es soll ein Einblick gegeben werden, warum und wie die Ziele umgesetzt wurden und welche Probleme bei der Umsetzung aufgetreten sind.

## Das Modellieren realistischer, variantenreicher Szenarien

In diesem Sprintziel wurde das Ausgangsbeispiel der Canstatter Wasen wieder aufgegriffen. Während im ersten Sprint nur selbst erdachte Szenarien erstellt wurden, wurden die Szenarien in diesem Sprint zum Teil von der Wasen inspiriert, um den Personenstrom an realistischen Orten zu simulieren. Dazu wurden einzelne möglicherweise kritische Schauplätze der Wasen von uns herausgegriffen und mittels VADERE nachgestellt.



Abbildung 2

This picture shows a site map of the festival „Canstatter Wasen“. Some of our created scenarios are inspired of this festival. Therefore we searched for points at this festival, which could get critical in case that a lot of person walk there.

Diese kritischen Schauplätze lagen unserer Meinung nach vor allem an den Punkten, an denen viele Menschen von unterschiedlichen Seiten kommen bzw. unterschiedlichste Wege nehmen konnten, sei es, weil sie von oder zu der Bahn, dem Parkplatz oder auch den Toiletten kamen oder gingen, die sich zum Teil am Rand befinden. Auch lange Gänge mit Hindernissen waren hier von Interesse, weil die Personen diesen auf ihrem Weg über das Festgelände ausweichen mussten.



Abbildung 3

This picture shows the site map of the festival „Canstatter Wasen“ too but in this one our points of interest are marked.

Die in der Abbildung markierten Bereiche stellen unsere kritischen Schauplätze dar, die wir als Szenarios in VADERE realisiert haben, um anschließend eine genauere Analyse durchführen zu können. Im Folgenden werden die einzelnen kritischen Plätze genauer betrachtet, bevor noch die Szenarios vorgestellt werden, die in Absprache mit den anderen beiden Gruppen in dem zweiten Sprint entstanden sind.



Abbildung 4

This picture shows the red marked place in the previous picture in detail. The green and orange rectangles show the excerpts of interest, which were realized as scenarios.

Das grüne und der orangene Ausschnitt von Kreuzungen auf der Wasen haben wir in 2 Szenarios aufgeteilt. Dieser Bereich war für uns deshalb von Bedeutung, weil die Personen hier von verschiedenen Seiten kommen konnten bzw. verschiedene Wege nehmen konnten, um diesen Bereich zu verlassen, denn oberhalb dieses Ausschnittes liegt der Parkplatz, auf der rechten Seite liegt die Bahnstation und die Toiletten und unterhalb erstreckt sich das weitere Festgelände.

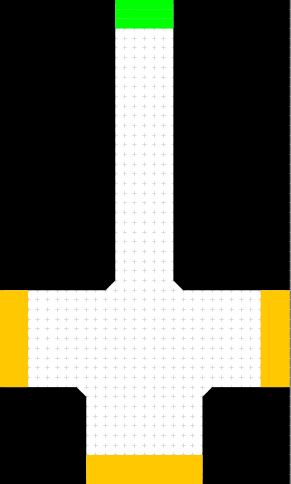
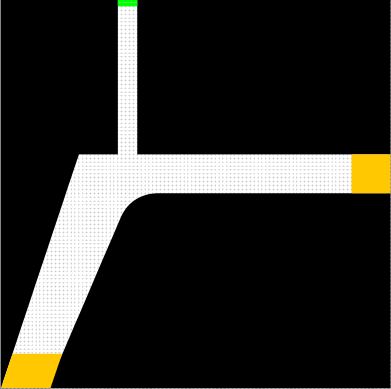


Abbildung 5

The implemented scenarios of the „W“sen“.

Bei der Realisierung dieser Ausschnitte als Szenarios ergab sich das Problem der abgerundeten Ecken auf dem Gelände, die wir aufgrund der realitätsnähe unbedingt in VADERE ebenfalls so umsetzten wollten. Ein Beispiel dafür ist das Festzelt in ganz linken grünen Szenario in Abbildung 4.

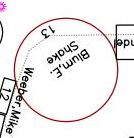
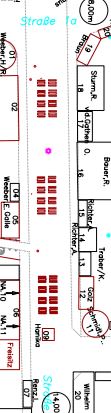


Abbildung 6

These pictures show an example of the round corners at the festival and our implementation in VADERE

Das Problem bei der Umsetzung dieser Rundungen in VADERE war, dass es in VADERE keine Möglichkeit gibt, runde Objekte in das Szenario einzufügen. Diese Rundungen mussten in VADERE angenähert werden.



Dieser Bereich der Wasen wurde ebenfalls als Szenario realisiert. Interessant waren hierbei zum einen die Hindernisse (blau markiert), denen die Menschen auf ihrem Weg über die Wasen ausweichen mussten, zum anderen die Abzweigungen an den Seiten der großen Hauptstraße (grün markiert) , die die Menschen nehmen konnten bzw. aus denen sie auf die Hauptstraße einbiegen konnten.

Abbildung 7

This excerpt of the site map of the festival shows the blue marked area in the second picture in detail. The green rectangles mark the branches, that depart from the main road.

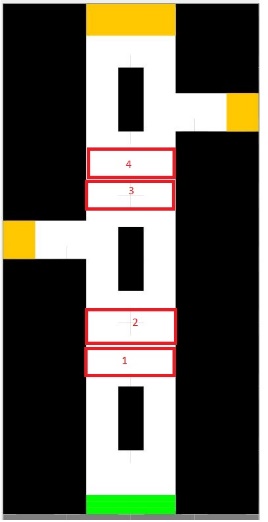


Abbildung 8

This picture shows the realization of the area above in VADERE with all camera sections (red rectangles).

Die weiteren Szenarien, die in diesem Sprint erstellt wurden, entstanden aufgrund der Wünsche der anderen beiden Gruppen. Auch diese sollen im Folgenden kurz gezeigt werden:

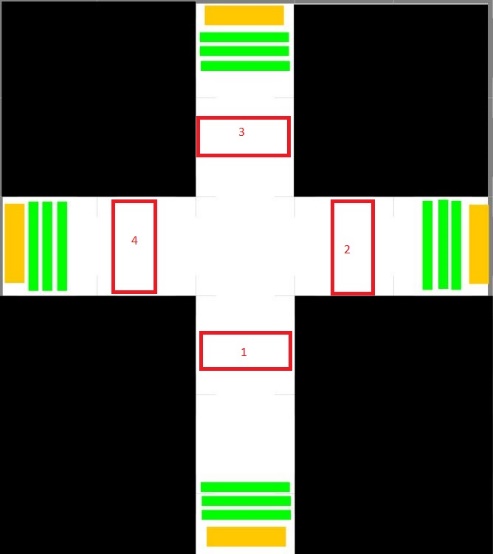


Abbildung 9

This picture shows one of our realized scenarios, which the other groups had wanted. The red rectangles mark the camera sections.

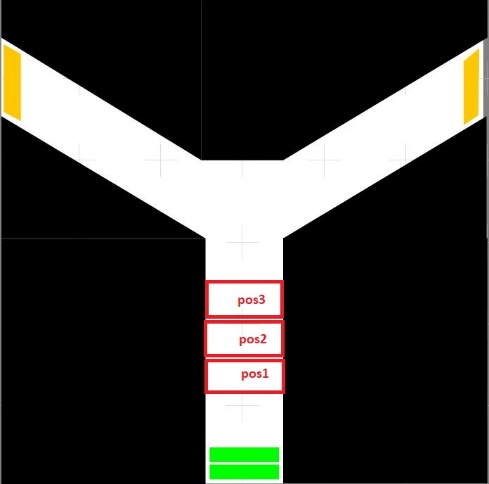
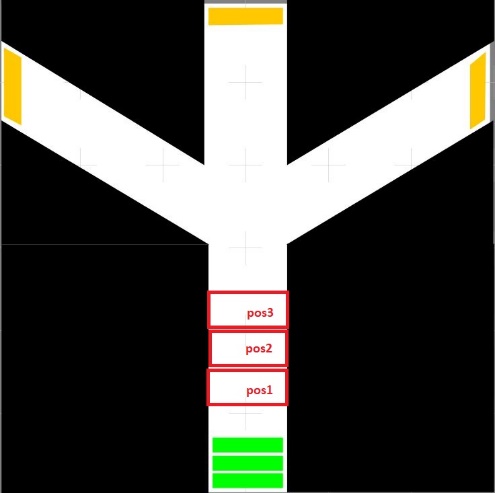


Abbildung 11

This scenario was also realized at the request of the other groups. Again, the red rectangles show the camera sections.

Abbildung 10

This picture also shows one of the realized scenarios, which the groups of the machinelearning algorithms had wanted. The red rectangles mark the camera sections.

## Die Filter in der Datenvorverarbeitung

Die bereits im ersten Sprint begonnene Literaturrecherche über die Filter in der Datenvorverarbeitung sollten in diesem Sprint weiter ausgebaut und auch einige ausgewählte Filter implementiert werden. Wie bereits erwähnt, bestand die Hoffnung hierbei darin, dass mittels einiger verschiedener Filter, unsere Daten so aufgewertet werden können, dass das Lernen mittels der Algorithmen des Maschinellen Lernens besser funktioniert.

Ein Problem bestand hierbei darin, dass es wenige Informationen zu den Filtern in der Vorverarbeitung für Neuronale Netze zu finden gab, weshalb sich hier an den Bildfiltern orientiert wurde. Ins Auge gestochen ist dabei zum einen der PCA-Filter (**P**rincipal **C**omponent **A**nalysis zu Deutsch Hauptkomponentenanalyse). Die Hauptkomponentenanalyse ist eine Standardmethode um umfangreiche Datensätze zu strukturieren, zu vereinfachen und zu veranschaulichen.[[1]](#footnote-1) Es gibt verschiedene Varianten die Hauptkomponentenanalyse durchzuführen. Zum einen kann dies über die Berechnung der Eigenvektoren von der Kovarianz Matrix der ursprünglichen Matrix durchgeführt werden. Zum anderen kann die Berechnung auch über eine SVD (**S**ingular **V**alue **D**ecomposition zu Deutsch Singulärwertzerlegung) durchgeführt werden. Da beide Methoden zu dem gewünschten Ergebnis führen und es für die SVD eine gut geeignete Methode gibt, wurde diese für die Implementierung gewählt. Die folgenden Erläuterungen sollen daher einen Einblick in die Umsetzung geben, die wir durchgeführt haben, um die effektive Dimension unserer Daten und damit auch die Dimension unseres Problems zu verringern­[[2]](#footnote-2).

Mathematisch gesehen ist die Hauptkomponentenanalyse eine Methode der Statistik, mit derer es möglich ist, in einen mehrdimensionalen Datensatz eine Transformation auf dessen Hauptachsen vorzunehmen. Dabei wird eine Matrix durch ihre Linearkombination dargestellt, sodass durch diese Projektion möglichst wenige Informationen verloren gehen[[3]](#footnote-3). Ausgangspunkt dieser Hauptkomponentenanalyse bildet bei uns unsere selbst berechnete Dichtematrix.

Die Singulärwerte einer Matrix sind die Quadratwurzeln der Eigenwerte von oder . Mittels einer Singulärwertzerlegung kann aus unserer Dichtematrix die folgende Zerlegung berechnet werden:

.

Bei einer Matrix enthalten wir hierbei die Matrix mit den linken Singulärvektoren, die Matrix mit den rechten Singulärvektoren und die Matrix welche auf ihrer Diagonalen die Singulärwerte enthält. Die Singulärwerte welche die Diagonalelemente von sind, sind der Größe nach sortiert, wobei gilt . Grund dafür ist, dass größere Singulärwerte als wichtiger zu erachten sind. Die Singulärvektoren, sind entsprechend ihrer Singulärwerte in den Matrizen und sortiert.

Da die kleinsten Singulärwerte kaum, oder Singulärwerte gleich Null sogar keinen, Einfluss auf die Daten haben, kann die Ausgangsmatrix einzig aus den größten Singuläwerten, ohne große Informationsverluste, wieder zusammengesetzt werden. Dazu kann man jedem Singulärwert einen Anteil in Prozent an der Totalvariation des Originalbildes zuordnen[[4]](#footnote-5). Dieser ist:

Man kann nun einen prozentualen Anteil festlegen den man von der Originalmatrix behalten will. Welche Singulärwerte als „zu klein“ gelten, um einen Einfluss auf die Daten zu haben, kann nun mittels dem zuvor festgelegten , welcher zwischen 0 und 100 liegt, gewählt werden sodass ein der minimale bestimmt werden kann, für welchen gilt:

wobei m und n die Dimensionen der Originalmatrix sind.[[5]](#footnote-6)

Wenn die zerlegten Matrizen nun bei abgeschnitten werden bleibt immer noch der Anteil an Informationen der Originalmatrix übrig. Die reduzierten Matrizen haben dann die Dimensionen:

Es ist jedoch ausreichend statt der kompletten Matrix nur die Diagonalelemente dieser zu speichern, wodurch die Dimension dieser sogar auf reduziert werden kann. Da in den meisten Fällen der größte Anteil der Totalvariation der Originalmatrix bereits durch sehr wenige Singulärwerte und damit mit einem kleinen dargestellt werden kann, kann hierdurch eine große Reduktion des benötigten Speichers erreicht werden.

Es wird nun eine Matrix betrachtet, die aus der Berechnung der Dichtewerte entstanden ist. Wenn diese als Bild ausgegeben wird, sieht sie folgendermaßen aus:



Abbildung 12

Original matrix (example)

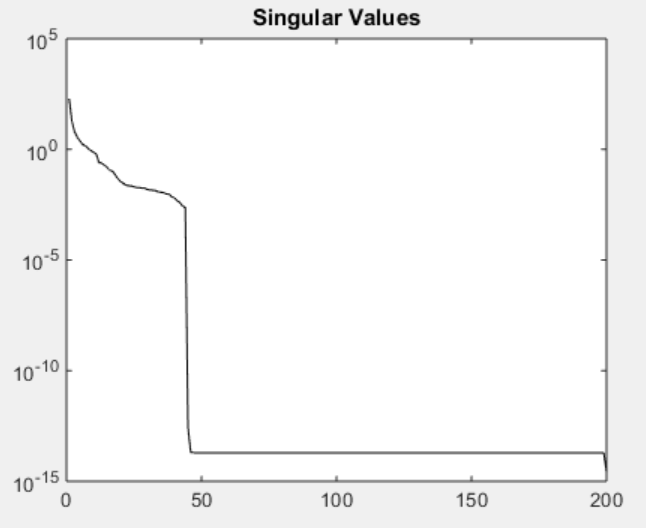


Abbildung 13

Singular values of the previous example matrix.

Betrachtet man nun die Singulärwerte des Beispielszenarios, wird deutlich, dass es nur sehr wenige große und damit beachtenswerte Singulärwerte gibt und die anderen abgeschnitten werden können.

Bei Betrachten des Bildes fällt sofort der extreme Abfall der Größe der Singulärwerte ca. beim 45. Wert auf. Dadurch weiß man bereits, dass man höchstens die ersten 45 Singulärwerte benötigen wird um die Informationen der Matrix zu erhalten.

Setzt man nun und berechnet somit die Menge der benötigten Singulärwerte um 99 % der Informationen der Originalmatrix zu behalten erhält man als Ergebnis sogar nur . Zu Testzwecken wurde die Matrix nun mit nur 10 Singulärwerten wieder zusammengesetzt wird, diese sieht folgendermaßen aus:



Abbildung 14

Reduced matrix with 99% of the original information

Mit bloßem Auge ist kein Unterschied zwischen den beiden Matrizen zu erkennen. Die durchschnittliche quadratische Abweichung zwischen beiden liegt auch nur bei .

Zum Vergleich wurde die Matrix noch mit einem betrachtet. Hier kann man gut die immer größer werdenden Unterschiede zur Originalmatrix sehen und damit deutlich den Informationsverlust nachvollziehen.



Abbildung 15

Reduced matrices with 95%, 90%, 75%, 50% of the original information.

Die Menge der verwendeten Singulärwerte (SV) und die durchschnittliche quadratische Abweichung (MSE) dieser reduzierten Matrizen sind Folgende:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 95% | 90% | 75% | 50% |
| Menge SV: | 6 | 4 | 2 | 1 |
| MSE: |  |  |  |  |

Tabelle 2

The table shows the amount of singular values used and the mean squared errors of these reduced matrices.

Nach Betrachten von 188 Szenarien wurde zur Darstellung von 99% der Orginalinformation höchstens ein benötigt.

Bei der Hauptkomponentenanalyse mithilfe der Singulärwertzerlegung ist es üblich die Matrix vor der Zerlegung zu zentrieren indem von jedem Wert einer Spalte der Mittelwert dieser abgezogen wird[[6]](#footnote-7). Wenn dies mit auf das obige Beispiel angewandt wird, sieht die wieder zusammengesetzte Matrix auf den ersten Blick jedoch nicht besonders viel versprechend aus. Dieser Ansatz sollte jedoch überprüft werden.



Abbildung 16

Reduced matrix with previous centering and with 99% of the original information

Des Weiteren wurde der Versuch unternommen, den Mittelwert aller Matrizen eines kompletten Durchlaufes von der Matrix abzuziehen. Hierbei werden die am häufigsten belaufenen Routen heller.



Abbildung 17

Reduced matrix after previous subtraction of the mean and with 99% of the original information.

Ein weiterer Filter, der in diesem Sprint umgesetzt wurde, war der Binärisierungsfilter, der zur Verbesserung der der Dichtedaten beitragen sollte. Die Bilder, die aus den Daten der Dichte erstellt worden sind, waren zum Teil sehr verschwommen, was vermutlich die Gruppen des Maschinellen Lernens am effektiven Lernen hindert. Unsere Hoffnung bestand nun darin, die Daten durch den Filter so zu verbessern, dass diese Verbesserung auch an den durch die Daten erzeugten Bilder zu sehen ist, mit dem Ziel, dass somit das Lernen mittels der Daten besser funktioniert. Dabei wurde sich für diesen Filter entschieden, weil dessen Hauptaufgabe die Verdeutlichung der Objekte, in unserem Fall die Dichte, ist.[[7]](#footnote-8)

Die Funktionsweise des Filters ist die folgende, zunächst wird eine Segmentierung der Daten vorgenommen, das heißt die Daten werden in einzelne Objekte zerlegt. Durch diese Segmentierung werden gewünschte Merkmale hervorgehoben und der Rest ignoriert. Ein einfaches Verfahren ist dafür das sogenannte Thresholding, also eine Binärisierung des Bildes. Bei diesem werden Pixelwerte von einem eindimensionalen Bild mit einem Schwellwert verglichen, wodurch die Zugehörigkeit des Pixels zum Objekt überprüft wird.

Dieser Filter wurde dabei programmatisch wie folgt umgesetzt: Die Dichtedaten eines Szenarios werden in Form einer Liste von Matrizen an den Filter übergeben. Dort werden die Werte zuerst normalisiert, sprich auf einen Wert zwischen 0 und 1 gebracht. Daraufhin werden die einzelnen Werte der Matrizen, sprich die Dichtewerte der Pixel durchlaufen und verglichen mit einem Schwellenwert. Das entsprechende Pixel wird dann auf 0 gesetzt, falls der Wert der Dichte darunter liegt, bzw. auf 1, also den Maximalwert, falls er darüber liegt.

## Die Verteilung der Personen in VADERE

Wie bereits erwähnt benötigten die Gruppen des Maschinellen Lernens eine enorme Menge an Daten, deren Erzeugung in VADERE per Hand sehr aufwendig ist. Das liegt daran, dass sowohl die Random-Forrest Gruppe, als auch die Neuronale Netze Gruppe, 60 unterschiedliche Verteilungen ein und desselben Szenarios benötigten, damit die jeweiligen Algorithmen effektiv lernen können. Die unterschiedlichen Verteilungen meinen in diesem Zusammenhang, dass die Anzahl an Personen, die in einem Szenario laufen, in unterschiedlicher Anzahl auf die einzelnen Ziele verteilt werden sollten.

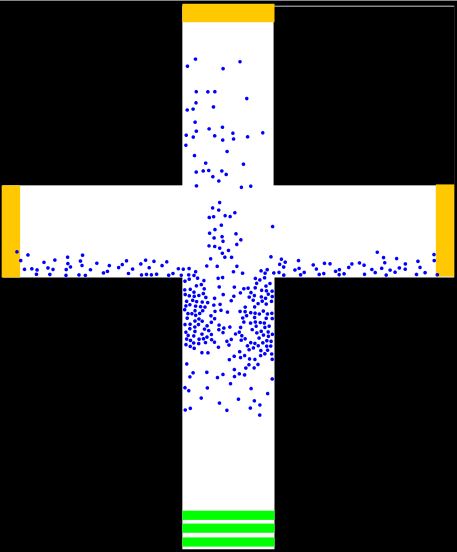


Abbildung 18

This picture shows an excerpt of one of our scenarios in VADERE. You can see some agents (blue dots) on their way to their respective targets (yellow rectangles). In this example, the agents are distributed equally among the three targest.

Zum Ende des ersten Sprints wurde diese Verteilung zufällig berechnet. Da bei dieser Variante der Berechnung der Verteilung das Problem von doppelt berechneten oder fehlenden Verteilungen bestand, war das Ziel des zweiten Sprints, die Verteilungen mittels Permutationen zu berechnen.

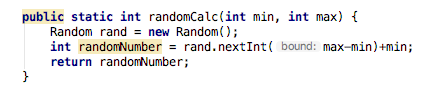
****

Abbildung 19

This short code snippet shows the calculation of the random distribution.

Alle möglichen Verteilungen wurden dazu in Python, mit der dort bereits vorhandenen Bibliothek „Itertools.permutations“ berechnet. Dies wurde wie folgt umgesetzt, es wurden zwei Listen mit den Werten 0-100 erstellt und in 5er Schritten durchlaufen. In jedem dieser Schritte wurde ein 3er- Tupel aufgestellt, dass die Verteilung der Personen auf die drei Ziele darstellen sollte. Dieses Tupel wurde durch Permutation mit Werten aus den beiden Listen gefüllt. Fehlende Verteilungen wurden somit vermieden und doppelte Verteilungen wurden nachträglich aussortiert.

Die gültigen Verteilungen mussten in Summe 100 ergeben, denn die einzelnen Elemente der Tupel geben den Prozentwert an, wie sich die Menschen auf die Ziele verteilen.



Abbildung 20

This short code snipped shows our implementation of all permutations.

Diese Datei wurde dann an unser bestehendes Java Programm, welches die entsprechenden Parameter während Simulation des Szenarios manipuliert, übergeben.

Zudem musste die Automatisierung an die entsprechenden Szenarien angepasst werden, denn zum Ende des ersten Sprints, konnten, aufgrund der Absprache mit den anderen beiden Gruppen, nur 3 Quellen in der Automatisierung simuliert werden und jetzt, zum Ende des zweiten Sprints, können Szenarien mit beliebig vielen Quellen simuliert werden. Bei der Automatisierung muss daher zu Beginn geprüft werden, wie viele Quellen vorhanden sind.

Da das VADERE Projekt mittels Multithreading arbeitet, bestand bei der Umsetzung dieses Ziels das Problem, dass die gewünschten 60 Verteilungen nicht gleichzeitig simuliert werden konnten, ohne, dass der Computer bei der Bewältigung dieser Arbeit streikte. Umgangen haben wir dieses Problem, in dem wir nun immer 4 (aufgrund der 4 Kerne unseres Computers) Verteilungen gleichzeitig simulieren lassen.

Natürlich wurde diese Automatisierung mit Hilfe von verschiedenen Tests überprüft. Kontrolliert wurde dabei ob beispielsweise die Anzahl der Quellen und Ziele in jeder Durchführung einer bestimmten Verteilung eines Szenarios übereinstimmen. Weiterhin wurde überprüft, ob die jeweiligen ID’s der Quellen und Ziele richtig übergeben werden und natürlich, ob letztendlich nach Durchführung der Simulationen die Anzahl an ausgeführten Verteilungen auch mit der Anzahl an Verteilungen übereinstimmt, die übergeben worden ist.

## Die Datenformate und die Datengenerierung

Natürlich bestand ein weiteres Ziel, wie auch schon im ersten Sprint, darin, Daten für die beiden Gruppen des Maschinellen Lernens zu generieren. Um dieses Ziel umzusetzen mussten zunächst aus den Ausgaben aus VADERE, d.h. aus den verschiedenen Ausgaben, die wir durch die Simulation sämtlicher benötigten Verteilungen in VADERE erhalten hatten, Dichte-Daten generiert und die Trajektorien formatiert werden. Da von den Gruppen des Maschinenlernalgorithmus so gewünscht, haben wir in diesem Sprint sogenannten „Experimentdaten“ erstellt, die den Namen tragen, weil bei der Erzeugung dieser Daten etwas experimentiert wurde. Dazu wurde die „Kamerahöhe“ bzw. die „Kamerabreite“ etwas verändert, die framerate heruntergesetzt und es wurde zu Beginn und am Ende keine Simulationszeit abgeschnitten, wie es sonst der Fall ist.

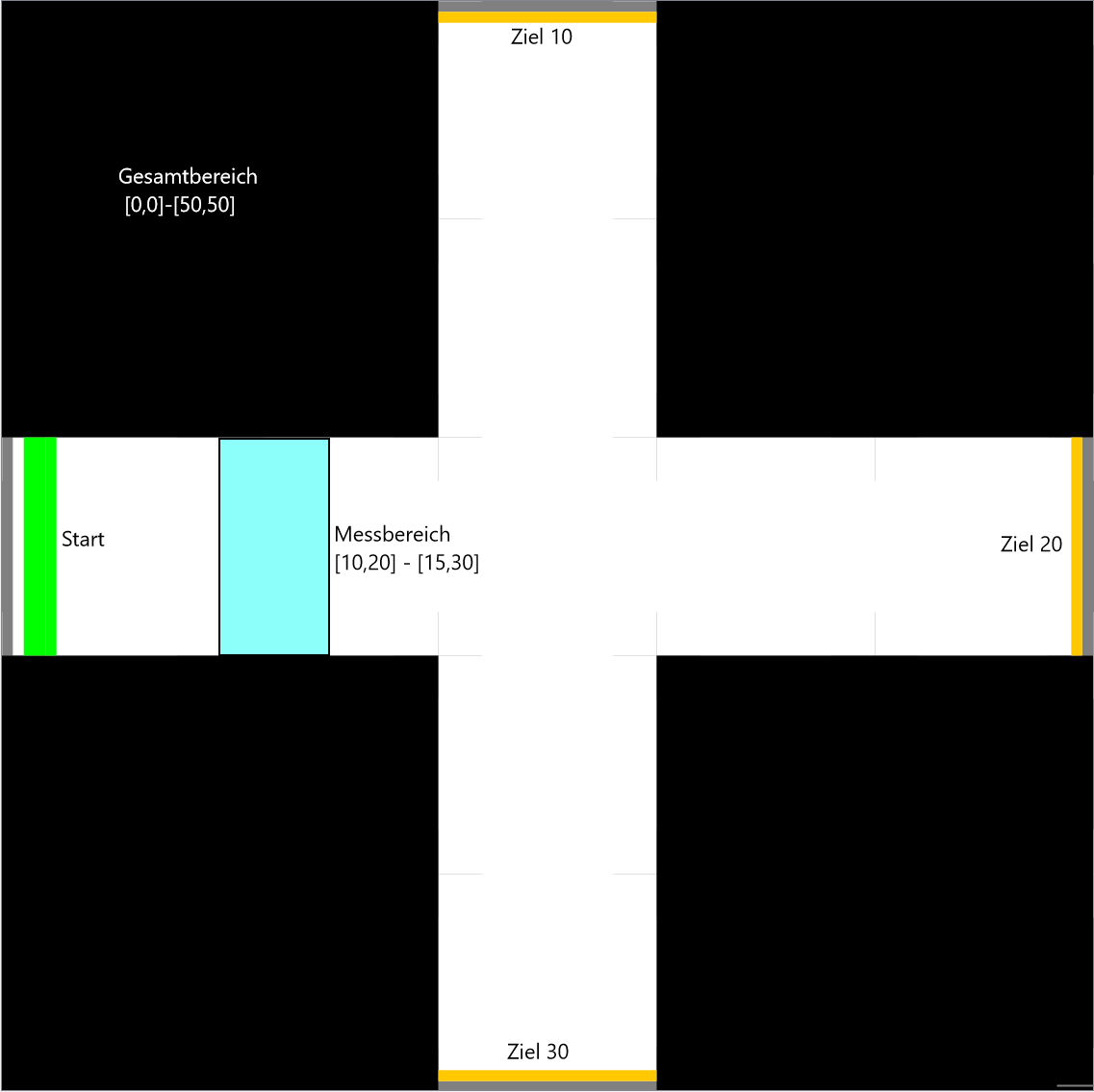
****

Abbildung 21

This picture shows one of our scenarios and the selected camera section (blue marked area).

Die bis jetzt erstellten Daten waren von der Simulationszeit immer relativ kurz, da pro Simulationsdurchlauf die gleiche Verteilung verwendet wurde. Die einzelnen csv-Dateien mit den verschiedenen Verteilungen konnten von den Machinlearning Gruppen zu einem großen Datensatz zusammengefügt werden, je nachdem, wie sie die Auswahl der Daten zusammenlegen wollten. Diese zusammengefügten Daten verfügten jedoch nicht mehr über einen kontinuierlichen Zeitverlauf.

Da die Neuronale Netzte Gruppe ihr Netzwerk anhand von Zeitverlaufsdaten trainieren können, musste ein weiterer Datensatz erstellt werden, bei dem dies geben ist.

.

Die Simulationszeit wurde hierfür auf 2000 Zeitschritte gesetzt und 3 verschiedene Simulationsdurchläufe mit je drei unterschiedlichen Kamerapositionen gemacht. Damit trotzdem die momentane Verteilung pro Zeitschritt verschieden ist, musste in VADERE die Personen Zufallsverteilt auf die Ziele (Targets) verteilt werden. Um dies umzusetzen, wurde der Code VADERES, mit Hilfe von Hernn Zönnchen, dahingehend geändert, dass sich die Personen zufällig auf die einzelnen Ziele aufteilen, somit wird nur noch eine Quelle benötigt, der alle Ziele zugeteilt werden können.

Weiterhin sollte in diesem Sprint das Format der Trajektorien dahingehend angepasst werden, dass alle Personen immer die gleiche Anzahl an Schritten aufweisen. Somit sollten die „-1“ Einträge verhindert werden, die im vorherigen Format anzeigten, dass eine Person in einem bestimmten Zeitschritt bereits im Ziel angekommen war, noch gar nicht die Quelle verlassen hatte, oder sich nicht in dem Kameraausschnitt befand. Grund für diese Änderung des Formats war, dass die Gruppe, die mit dem Neuronalen Netz arbeitet, Probleme mit der Anzahl an „-1“ – Werten in der Matrix hatte.

Deshalb wurde nach den Vorgaben der Machinelearning Gruppen ein neues Format festgelegt.

Für dieses wurde der Kameraausschnitt in Intervalle unterteilt. Als Intervall Größe wurde der Größe Schritt aller Personen genommen. Pro Intervall wird immer nur ein Schritt jeder Person übernommen. Hierdurch wird sichergestellt, dass alle Personen die gleiche Anzahl an Schritten haben und dadurch jede Zeile in der später ausgegebenen Matrix gleich lang ist.

Pro csv-Datei ist jedoch die Anzahl der Schritte bzw. Intervalle immer noch verschieden, da diese immer von dem größten Schritt aller Personen abhängig ist. Deshalb wurde geprüft was die kleinste Anzahl an Intervallen war und diese wurde einmal global verwendet.

Zu beachten ist, dass durch diese Verarbeitung der Trajektorien, die zeitliche Abhängigkeit der Daten verloren geht. Beispielsweise gibt es somit keine doppelten x und y Positionen mehr, was eine wartende Person beschrieben hätte.

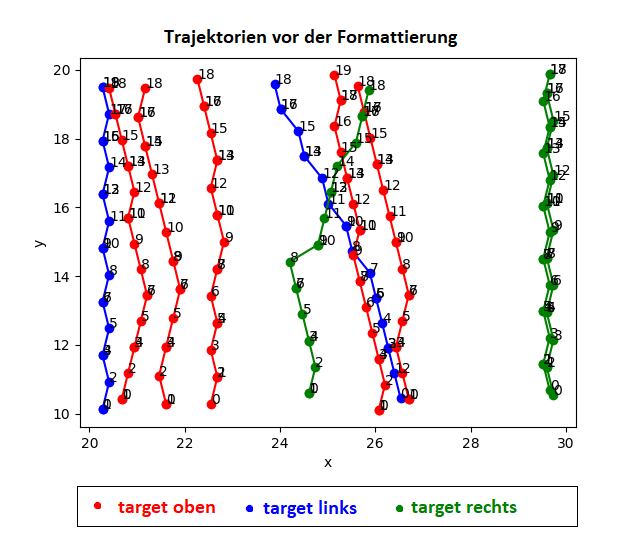


Abbildung 22

This picture shows the amount steps two different people needed to go to the target.

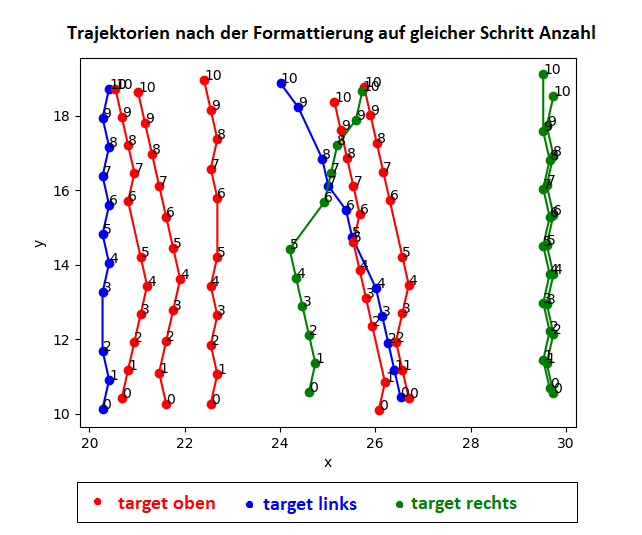


Abbildung 23

This picture shows the amoutn of steps each person needs to go to the target (after formatting).

Ein vorgegebenes Ziel war die Darstellung der Daten in dem Datenformat des Fraunhofer Instituts. Dieses Format hat die folgende Form:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Dichtematrix | Geschwindigkeitsmatrix | Momentane Zielverteilung |
| Zeitschritt 1 | [[p, p, …],[p, p, ...], …] | [[v, v, …],[v, v, ...], …] | (20, 30, 50) |
| Zeitschritt 2 | … | … | … |

Tabelle 3

The table shows the data format, which is used by the Fraunhofer Institut.

Um das Fraunhofer Format umzusetzen, wird das bestehende Format der Dichtedaten wie in der obigen Tabelle erweitert. Die Berechnung der Geschwindigkeitsmatrix wird im Folgenden erklärt.

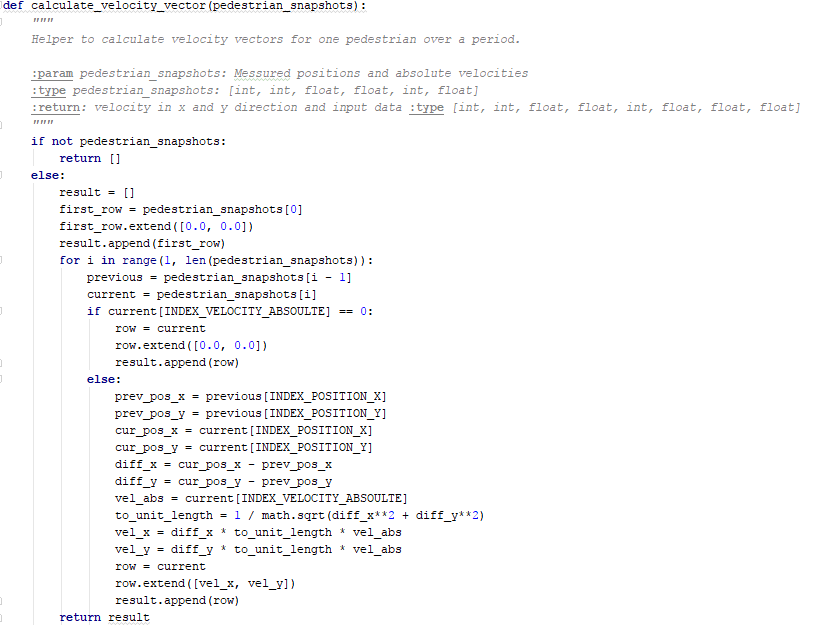
Vadere liefert durch den Output-Prozessor „VelocityProcessor“ absolute Geschwindigkeiten, die für jeden Zeitschritt pro Fußgänger vorliegen. Aus der absoluten Geschwindigkeit und der Positionsänderung eines Fußgängers zwischen zwei Zeitschritten (Richtung), wird eine vektorielle Geschwindigkeit errechnet. Nun wird das Beobachtungsfenster in ein Grid unterteilt. Für jedes Grid-Element, dessen Größe durch eine angegebene Auflösung bestimmt ist, wird die gemittelte Geschwindigkeit aller enthaltenen Fußgänger berechnet. Diese Berechnung findet für jeden Simulationszeitschritt statt.

Abbildung 24

This code snippet shows the calculation of the velocity vector.

## Das Testen unserer Software

Ein weiteres Ziel dieses Sprints war das Testen unserer Implementierungen, um sicherzustellen, dass diese fehlerfrei funktionieren. In diesem Rahmen wurden von uns in diesem Sprint Dichte-Tests durchgeführt. Da wir für jeden Schritt die Bilder aus unseren Dichte-Daten generieren, entstand, für jedes Szenario, mit dessen Hilfe Dichte-Daten berechnet wurden, eine Reihe von Bildern der Dichte. Aus diesen Bildern wurde dann ein Video erstellt, welches mit dem Video der Postvisualisierung in VADERE verglichen und auf Korrektheit hin überprüft werden konnte.

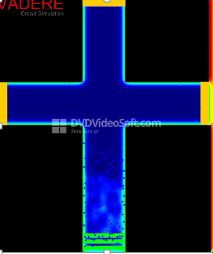
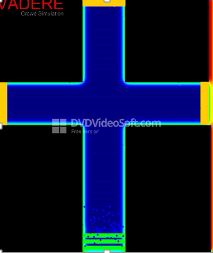


Abbildung 25

Graphical representation of the density.

Abbildung 26

Graphical representation of the density in another step.

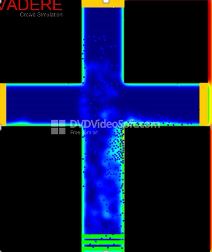


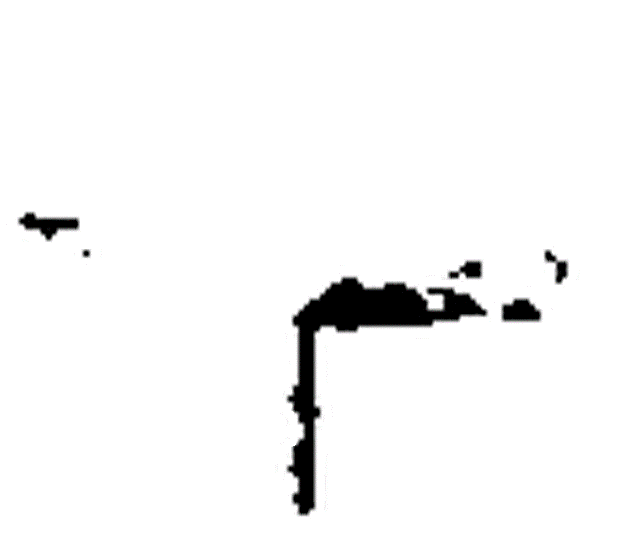
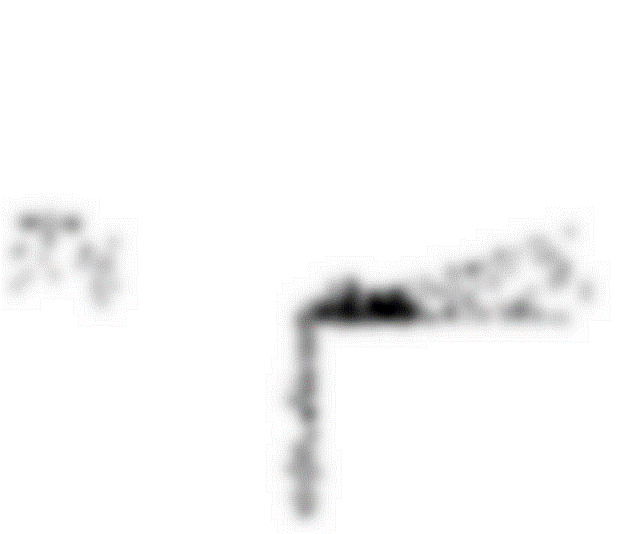
Abbildung 27

Graphical representation of the density.

Auch die Filter wurden auf ihre korrekte Arbeitsweise hin überprüft. Beispielsweise wurden für den Binärisierungsfilter ebenfalls die Bilder der Dichtedaten mit den Bildern verglichen, die durch die gefilterten Daten entstanden sind. Somit konnte überprüft werden, dass die Daten nur verbessert, nicht aber in ihrem Aufbau verändert wurden.

Abbildung 28

On the left hand side you can see the graphical representation of the density (not edited by the filter) and on the right hand side you can see the same graphical representation of the density but this time edited by the filter.



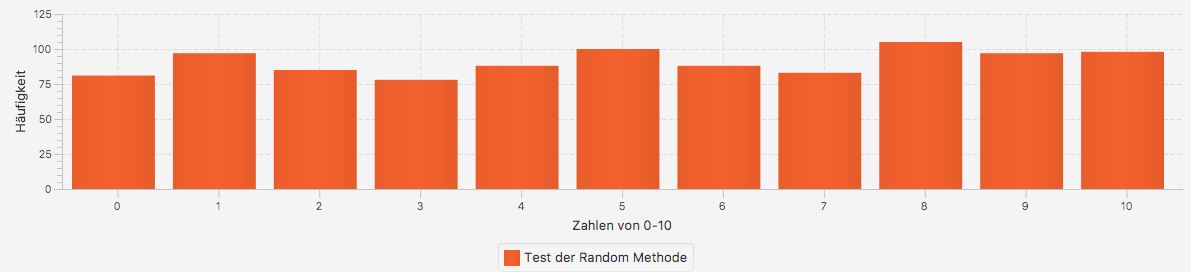
Zu Beginn des Sprints wurde zu dem noch getestet, ob die zufällig berechnete Verteilung bei der Automatisierung von VADERE uniform verteilt ist, d.h. ob wirklich alle Zahlen gleich oft vorkommen. Dazu wurde die entsprechende Methode 1000 aufgerufen und deren Ausgaben in folgendem Histogramm festgehalten.

Abbildung 29

The figure shows the histogram of the numbers calculated for the random distribution

Des Weiteren wurde mit Hilfe von Unit Tests unser bisheriges Programm zu testen.

## Optimierung und Dokumentation des bisherigen Codes

Ein weiteres, wichtiges Sprintziel des zweiten Sprints war die Optimierung und Dokumentation unseres bisherigen Codes. Verschiedene Dinge wurden dabei umgesetzt:

* Das Einlesen der verschiedenen Dateien wurde verbessert. Während im ersten Sprint die Indizes der Zeilen in der jeweiligen Datei noch fest codiert waren, werden nun die
* Alle Funktionen, die sich mit dem Einlesen von Daten beschäftigen, sind nun in einer eigenen Klasse „FileReader“. Im Laufe des 2. Sprints wurden verschiede Datensätze erstellt. Dabei unterschieden sich die Datensätze sowohl in der Menge der simulierten Fußgänger und in der Verteilung der Fußgänger auf die Ziele als auch durch die simulierten Szenarien.

Die FileReader-Klasse erkennt alle Input-Dateien eines zu generierenden Datensatzes und gibt diese Information weiter an das „FileWriter“-Modul.

* Alle Funktionen, die sich mit dem Schreiben von Daten beschäftigen, sind nun ebenfalls in einer eigenen Klasse namens „FileWriter“. Das Modul erstellt für jeden Datensatz ein entsprechendes Verzeichnis und ermöglicht dadurch eine Abgrenzung der Datensätze untereinander und eine eindeutige Zuordnung der Datensätze zu seinen Input-Dateien.
* Programmcode, der sich wiederholt oder für mehrere Module brauchbar ist, wurde in das util package verschoben. Eine wiederkehrende Aufgabe ist das Selektieren, das Gruppieren und das Sortieren der Daten. Hierfür wurden utility-Funktionen implementiert.
* Die Unterteilung der Anwendung in Module mit klaren Verantwortungsbereich erleichtert die Implementierung neuer Funktionen und Features als auch die gemeinsame Entwicklung der Anwendung. In der unten stehenden Abbildung ist die umgesetzte Struktur dargestellt.
* Der Programmcode wurde durch eine Dokumentation erweitert. Natürlich wird diese, aufgrund des sich möglicherweise noch ändernden Codes, auch im 3. und damit letzten Sprint weiter angepasst.

## Weitere Ziele des 2. Sprints

Zu den weiteren Zielen des zweiten Sprints zählten unter anderem die Aktualisierung des bestehenden Trello Projektes und unseres Repositories, sowie die weitere Organisation der Treffen, die natürlich auch während des zweiten Sprints stattgefunden haben. Um die gute Kommunikation des ersten Sprints weiterzuführen, wurden neben unseren wöchentlichen Treffen außerhalb der Vorlesung an den Wochenenden Skype-Meetings veranstaltet, um Probleme oder Änderungen der Aufgaben besprechen zu können. So wurde sichergestellt, dass kein Teammitglied mit möglicherweise entstehenden Problemen alleingelassen wurde.

# Implementierungen

Für unsere programmierarbeiten haben wir uns, wie bereits erwähnt, zu Beginn des ersten Sprints ein eigenes Repository zugelegt, in dem wir gemeinsam arbeiten konnten. Aufgrund der Größe, die unser Programm mittlerweile erreicht hat, möchten wir in diesem Abschnitt darauf verzichten, den Code hier einzufügen und stattdessen auf unser Repository verweisen, zu dem wir ihnen den Zugang gewähren.

<https://github.com/Datagroup-ModSim>

# Zusammenfassung und Ausblick in den 3. Sprint

Die Ziele, die wir uns für den zweiten Sprint vorgenommen hatten, konnten wir, trotz einiger Probleme, in diesem Sprint umsetzen. Für den dritten und damit letzten Sprint bleiben natürlich trotzdem noch einige Ziele übrig.

Neben weiterer Optimierung und Dokumentation unseres Programmcodes, sowie dem Testen unserer Implementierungen und dem Erstellen von Demonstrationen für die Firma accurate und den Studieninformationstag, ist ein weiteres Ziel die Anfertigung eines Posters für die PED 2018. Weiterhin soll der technische Bericht im 3. Sprint noch weiter ausgebaut werden.

Das große übergeordnete Ziel des dritten und letzten Sprints ist es jedoch, die Frage klären zu können, um die es in diesem Projektstudium von Anfang an ging: „Can we learn were people go?“

1. <http://homepages.physik.uni-muenchen.de/~f.abudinen/PCA.pdf> [↑](#footnote-ref-1)
2. http://www.statoek.wiso.uni-goettingen.de/veranstaltungen/Multivariate/Daten/mvsec4.pdf [↑](#footnote-ref-2)
3. <https://books.google.de/books?id=H0nSvM-DEqcC&printsec=frontcover&hl=de#v=onepage&q&f=false> [↑](#footnote-ref-3)
4. http://www.statoek.wiso.uni-goettingen.de/veranstaltungen/Multivariate/Daten/mvsec4.pdf [↑](#footnote-ref-5)
5. <https://books.google.de/books?id=H0nSvM-DEqcC&printsec=frontcover&hl=de#v=onepage&q&f=false> [↑](#footnote-ref-6)
6. https://www.mathworks.com/company/newsletters/articles/professor-svd.html [↑](#footnote-ref-7)
7. <http://www.christianbenjaminries.de/scientific/2010-04-13-Ries.pdf> [↑](#footnote-ref-8)