



Manuel Haid, BSc

# **Mobile Multisensorlösung zur Unterstützung von Sicherheits- und Risikoaufgaben bei Großveranstaltungen**

## **Masterarbeit**

to achieve the university degree of

Diplom-Ingenieur

Master's degree programme: Software-Entwicklung Wirtschaft

submitted to

**Graz University of Technology**

Supervisor

Dipl.-Ing. Dr.techn. Roman Kern

Institute of Interactive Systems and Data Science  
Head: Univ.-Prof. Dr. Stefanie Lindstaedt

Graz, September 2018

This document is set in Palatino, compiled with pdfL<sup>A</sup>T<sub>E</sub>X2e and Biber.

The L<sup>A</sup>T<sub>E</sub>X template from Karl Voit is based on KOMA script and can be found online: <https://github.com/novoid/LaTeX-KOMA-template>

## Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Das in TUGRAZonline hochgeladene Textdokument ist mit der vorliegenden Dissertation identisch.

---

Datum

---

Unterschrift

# Danksagung

# Abstract

Dramatic tragedies at major events in recent years with many deaths have shown how important it is to develop a security solution to prevent such catastrophes. In the context of this master thesis, a development concept for a mobile multisensor solution was developed, tested and evaluated to support safety and risk tasks at major events. After a detailed hardware research, a first prototype was developed, which was tested at the Frequency Festival in St. Pölten. The impressions and results from this test were evaluated and then a second prototype was developed, tested and subsequently evaluated. In addition to the detailed research of the various hardware components, Global Positioning System (GPS) and Inertial Measurement Unit (IMU) accuracy tests were conducted between professional sensors and smartphone sensors. Finally, a ready-to-use mobile multi-sensor solution was developed to support security and risk issues at major events designed to help security personnel in security tasks at urban locations and major events, thereby avoiding potentially dramatic tragedies.

# Kurzfassung

Dramatische Tragödien bei Großveranstaltungen in den letzten Jahren mit vielen Toten haben gezeigt, wie wichtig es ist eine Sicherheitslösung zu entwickeln um solche Katastrophen verhindern zu können. Im Rahmen dieser Masterarbeit wurde ein Entwicklungskonzept für eine Mobile Multi-sensorlösung zur Unterstützung von Sicherheits- und Risikoaufgaben bei Großveranstaltungen erarbeitet, getestet und evaluiert. Dabei wurde nach einer ausführlichen Hardwarerecherche ein erster Prototyp entwickelt, der danach am Frequency Festival in St. Pölten getestet wurde. Die Eindrücke und Ergebnisse aus diesem Test wurden ausgewertet und anschließend wurde ein zweiter Prototyp entwickelt, getestet und daraufhin evaluiert. Neben der ausführlichen Recherche der verschiedenen Hardwarekomponenten, wurden noch zusätzlich Global Positioning System (GPS)- und Inertial Measurement Unit (IMU) Genauigkeitstests zwischen professionellen Sensoren, sowie Smartphone Sensoren durchgeführt. Schlussendlich wurde eine fertige mobile Multisensorlösung zur Unterstützung von Sicherheits- und Risikoaufgaben bei Großveranstaltungen entwickelt, die Sicherheitsmitarbeiter bei Sicherungsaufgaben bei urbanen Plätzen und Großveranstaltungen unterstützen sollen und dadurch womögliche dramatische Tragödien verhindern können.

# Inhaltsverzeichnis

<b>Kurzfassung</b>	<b>vi</b>
<b>Abstract</b>	<b>v</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Hintergrund . . . . .	1
1.2 Problemstellung . . . . .	3
1.3 Rahmenbedingung . . . . .	3
1.4 Forschungsfragen . . . . .	4
<b>2 Grundlagen</b>	<b>6</b>
2.1 Smartphone . . . . .	6
2.2 Smartphone Betriebssysteme . . . . .	8
2.2.1 Android . . . . .	9
2.2.2 Architektur . . . . .	11
2.2.3 Android Java . . . . .	12
2.3 Einplatinencomputer . . . . .	12
2.4 Omnidirektionale Kamera . . . . .	12
2.5 Handheld Gimbal . . . . .	13
2.6 Webserver . . . . .	13
2.6.1 HTML . . . . .	14
2.6.2 CSS . . . . .	14
2.6.3 JavaScript . . . . .	14
2.7 C# . . . . .	14
2.8 Python . . . . .	15
2.9 Positionsbestimmungsverfahren . . . . .	15
2.9.1 GNSS . . . . .	16
2.9.2 RTK . . . . .	17

## Inhaltsverzeichnis

2.9.3	Inertial Measurement Unit (IMU) . . . . .	18
<b>3</b>	<b>Bestehende Systeme</b>	<b>21</b>
3.1	Bestehende Multi-Sensorlösungen . . . . .	21
3.1.1	Problemerkennung . . . . .	21
3.1.2	Problemweiterleitung bzw. Problemlösung . . . . .	22
3.2	Genauigkeitsmessverfahren . . . . .	23
3.2.1	GPS . . . . .	23
3.2.2	IMU . . . . .	24
<b>4</b>	<b>Methodik</b>	<b>30</b>
4.1	Prototyp 1 . . . . .	31
4.1.1	Hardwarerecherche . . . . .	31
4.1.2	Softwareimplementierung . . . . .	36
4.1.3	Tests am Frequency . . . . .	48
4.1.4	Evaluierung der Tests . . . . .	49
4.2	Prototyp 2 . . . . .	52
4.2.1	Hardwarerecherche . . . . .	52
4.2.2	Softwareimplementierung . . . . .	53
4.2.3	Tests . . . . .	54
4.2.4	Evaluierung der Tests . . . . .	55
<b>5</b>	<b>Ergebnis</b>	<b>70</b>
5.1	Vorbereitung . . . . .	75
5.2	Durchführung der Tests . . . . .	82
5.3	Nachbereitung . . . . .	84
5.4	Testergebnisse . . . . .	93
5.4.1	Positionsgenauigkeit Smartphone GPS Sensoren . . . . .	93
5.4.2	Positionsgenauigkeit Professionellen GPS Sensor . . . . .	96
5.4.3	Positionsgenauigkeit RTK System . . . . .	98
5.4.4	Professionellen GPS Sensor vs. Smartphone GPS Sensoren vs. RTK System . . . . .	99
5.4.5	Professionellen GPS Sensor vs. Smartphone Z5 Sensor vs. RTK Sensor Genauigkeitstest bei Bewegung . . . . .	102
5.4.6	Professionellen IMU Sensor vs. Smartphone IMU Sensoren vs. RTK System . . . . .	102
5.5	Diskussion . . . . .	103



## Inhaltsverzeichnis

<b>6 Schlussfolgerung</b>	<b>126</b>
<b>Literatur</b>	<b>131</b>

# Abbildungsverzeichnis

2.1	Anzahl Smartphone Nutzer weltweit . . . . .	6
2.2	Anzahl App Downloads weltweit . . . . .	7
2.3	Marktanteil Betriebssysteme . . . . .	8
2.4	Android Versionsverteilung . . . . .	10
2.5	Android Architektur . . . . .	19
2.6	RTK Technologie . . . . .	20
2.7	Inertial Measurement Unit (IMU) . . . . .	20
3.1	Positionsgenauigkeit der drei getesteten Smartphones . . . . .	26
3.2	Genauigkeit der verschiedenen App-Anwendungen . . . . .	27
3.3	Durschnittliche GPS-Genauigkeit verschiedener Android Geräten	28
3.4	Durschnittliche IMU-Genauigkeit verschiedener Android Geräten	28
3.5	IMU-Genauigkeit zwischen einzelmem IMU-Sensor und Kom- binationenlösung . . . . .	29
4.1	Zeitplan Masterarbeit . . . . .	30
4.2	Ricoh Theta R . . . . .	33
4.3	Raspberry Pi . . . . .	34
4.5	Osmo Mobile . . . . .	35
4.6	Sony Z5 . . . . .	36
4.7	Hardwarekonfiguration . . . . .	37
4.8	Android Versionsabdeckung . . . . .	38
4.9	Streamingkonfiguration . . . . .	43
4.10	Testseite Webserver . . . . .	44
4.11	Webseite vollständig . . . . .	57
4.12	Webseite Map . . . . .	58
4.13	Webseite Bild und Livestream . . . . .	59
4.14	Webseite Steuerungsmöglichkeiten . . . . .	59

## Abbildungsverzeichnis

4.15	Statusabfrage . . . . .	60
4.16	Containerbüro am Frequency . . . . .	60
4.17	Containerbüro Bildschirme am Frequency . . . . .	61
4.18	Masten am Quad montiert . . . . .	62
4.19	Hardwarekoffer 1. Prototyp . . . . .	63
4.20	Masten am Quad ausgezogen . . . . .	64
4.21	Masten im Einsatz . . . . .	65
4.22	Tablet App . . . . .	66
4.23	MutliSensor App . . . . .	66
4.24	Osmo Zenmuse X5R . . . . .	67
4.25	Emlid Reach RTK . . . . .	67
4.26	ReachView Update . . . . .	68
4.27	ReachView App . . . . .	69
4.28	DJI Akkustand . . . . .	69
5.1	Sony Z5 . . . . .	71
5.2	Galaxy S7 . . . . .	71
5.3	Nexus 6 . . . . .	73
5.4	Nexus 5 . . . . .	74
5.5	Moto G . . . . .	74
5.6	Android App für GPS Daten . . . . .	76
5.7	C# TCP Server für Positionsdatenempfang . . . . .	77
5.8	CSV Format am Handy . . . . .	77
5.9	u-Center . . . . .	78
5.10	RTK Settings . . . . .	79
5.11	RTK GNSS . . . . .	79
5.12	RTK Input . . . . .	80
5.13	RTK Output . . . . .	81
5.14	RTK App . . . . .	81
5.15	Maps Referenzpunkte . . . . .	83
5.16	Referenzpunkt B1 . . . . .	84
5.17	Referenzpunkt B1 mit Hardware . . . . .	84
5.18	Referenzpunkt PP2 . . . . .	85
5.19	Referenzpunkt PP2 mit Hardware . . . . .	85
5.20	Referenzpunkt PP24 . . . . .	86
5.21	Professionelles GPS . . . . .	86
5.22	Referenzpunkt PP232 . . . . .	87

## Abbildungsverzeichnis

5.23	RTK Basistation . . . . .	87
5.24	RTK Webseite . . . . .	88
5.25	GPS Beispielesdaten vom Referenzpunkt PP24 . . . . .	88
5.26	Koordinaten der Referenzpunkte . . . . .	89
5.27	GPS U-Blox Beispielesdaten . . . . .	90
5.28	GPS Visualizer <sup>19</sup> . . . . .	91
5.29	RTKCONV . . . . .	91
5.30	RTKCONV Optionen . . . . .	92
5.31	RTKPOST . . . . .	93
5.32	RTKPOST Einstellungen . . . . .	95
5.33	GPS Visualizer . . . . .	96
5.34	Positionsgenauigkeit Smartphones vom Referenzpunkt B1 . .	105
5.35	Positionsgenauigkeit Smartphones vom Referenzpunkt PP2 .	106
5.36	Positionsgenauigkeit Smartphones vom Referenzpunkt PP24	107
5.37	Positionsgenauigkeit Smartphones vom Referenzpunkt PP232	108
5.38	Positionsgenauigkeit GPS Gerät am Referenzpunkt B1 . . . .	109
5.39	Positionsgenauigkeit GPS Gerät am Referenzpunkt B1 . . . .	110
5.40	Positionsgenauigkeit GPS Gerät am Referenzpunkt B1 . . . .	111
5.41	Positionsgenauigkeit GPS Gerät am Referenzpunkt B1 . . . .	112
5.42	Positionsgenauigkeit RTK Emlid Reach am Referenzpunkt B1	113
5.43	Positionsgenauigkeit RTK Emlid Reach Roverstation am Referenzpunkt B1 . . . . .	114
5.44	Positionsgenauigkeit RTK Emlid Reach am Referenzpunkt PP2	115
5.45	Positionsgenauigkeit RTK Emlid Reach am Referenzpunkt PP2	116
5.46	Positionsgenauigkeit RTK Emlid Reach am Referenzpunkt PP24	117
5.47	Positionsgenauigkeit RTK Emlid Reach am Referenzpunkt PP24	118
5.48	Positionsgenauigkeit RTK Emlid Reach am Referenzpunkt PP232 . . . . .	119
5.49	Positionsgenauigkeit RTK Emlid Reach am Referenzpunkt PP232 . . . . .	120
5.50	Positionsgenauigkeit Sony Z5 vs. GPS u-blox am Referenzpunkt B1 . . . . .	121
5.51	Positionsgenauigkeit Sony Z5 vs. GPS u-blox am Referenzpunkt PP2 . . . . .	122
5.52	Positionsgenauigkeit Sony Z5 vs. GPS u-blox am Referenzpunkt PP24 . . . . .	123

## Abbildungsverzeichnis

5.53	Positionsgenauigkeit Sony Z5 vs. GPS u-blox am Referenzpunkt PP232 . . . . .	124
5.54	Positionsgenauigkeit u-blox GPS Sensor vs. Smartphone Z5 Sensor vs. RTK Sensor bei Bewegung . . . . .	125

# Tabellenverzeichnis

2.1	Android Versionen . . . . .	9
4.1	Raspberry Pi . . . . .	34
4.2	Sony Z5 . . . . .	36
4.3	Übersicht aller Kommandos . . . . .	47
5.1	Übersicht Positionsgenauigkeit der Smartphones vom Referenzpunkt B1 . . . . .	94
5.2	Übersicht Positionsgenauigkeit der Smartphones vom Referenzpunkt PP2 . . . . .	94
5.3	Übersicht Positionsgenauigkeit der Smartphones vom Referenzpunkt PP24 . . . . .	97
5.4	Übersicht Positionsgenauigkeit der Smartphones vom Referenzpunkt PP232 . . . . .	97
5.5	Übersicht Positionsgenauigkeit von Sony Z5 vs. GPS u-blox vs. RTK Rover vom Referenzpunkt B1 . . . . .	100
5.6	Übersicht Positionsgenauigkeit von Sony Z5 vs. GPS u-blox vs. RTK Rover vom Referenzpunkt PP2 . . . . .	100
5.7	Übersicht Positionsgenauigkeit von Sony Z5 vs. GPS u-blox vs. RTK Rover vom Referenzpunkt PP24 . . . . .	101
5.8	Übersicht Positionsgenauigkeit von Sony Z5 vs. GPS u-blox vs. RTK Rover vom Referenzpunkt PP232 . . . . .	101
5.9	Übersicht IMU Genauigkeitstest von Professionellen IMU Sensor vs. Smartphone IMU Sensoren vs. RTK System . . . .	103
5.10	Übersicht Endresultat von Sony Z5 vs. GPS u-blox vs. RTK Rover Positionsgenauigkeit von allen Referenzpunkten . . . .	104

# 1 Einleitung

## 1.1 Hintergrund

Dramatische Tragödien in den letzten Jahren, wie zum Beispiel der Panikausbruch auf der Stadiontribüne in Bradford 1985 mit 56 Toten, dem Feuer des Great-White Konzerts im Jahr 2003 auf Rhode Island mit mehr als 90 Toten oder auch die Vielzahl an Massenpaniken in Mekka mit mehreren tausenden Toten haben gezeigt, wie wichtig es ist eine Sicherheitslösung zu entwickeln um solche Katastrophen verhindern zu können [Zac14].

Weitere Beispiele sind Katastrophen beim Air & Style Event im Innsbrucker Bergisel-Stadion bei dem es 1999 fünf Tote gab oder auch die Massenpanik bei der Love Parade in Duisburg 2010 <sup>1</sup>.

Diese schlimmen Tragödien bei Großveranstaltungen zeigen auch auf, dass die Sicherung von Veranstaltungen mit einer großen Anzahl an Besuchern absolut wichtig und notwendig ist um solche Katastrophen verhindern zu können.

Nicht nur die Anzahl an Großveranstaltungen nehmen Jahr für Jahr zu, sondern auch die Menge der Besucher für jeder dieser Veranstaltungen. Damit diese auch ohne größere Probleme bewältigt werden können, muss das Event fehlerfrei organisiert und geplant werden. Jedoch aufgrund von dynamisch wechselnden Sicherheitslagen, sowie der erhöhten Effizienzsteigerung

---

<sup>1</sup>[https://diepresse.com/home/panorama/welt/583438/Chronologie\\_Tote-bei-Massenveranstaltungen-](https://diepresse.com/home/panorama/welt/583438/Chronologie_Tote-bei-Massenveranstaltungen-)

## 1 Einleitung

bei Personalressourcen, sind die Anforderungen an Sicherheitsdienste, Rettungskräfte, sowie Hilfsorganisationen enorm.

Diese Masterarbeit soll durch die Verwendung von schnellen, einfachen Kommunikationsmöglichkeiten, wie zum Beispiel Smartphones, einen Beitrag dazu leisten den zuständigen Organisationen bei der Sicherung von Großveranstaltungen bestmöglich zu unterstützen, damit diese Dienste den hohen Anforderungen gerecht werden können.

Der Schlüssel für eine optimale Einsatzführung und einer erfolgreichen Bewältigung von verschiedenen Gefährdungslagen besteht darin, dass kritische Situationen schnellst möglichst erfasst werden können und diese Informationen dann auch rasch innerhalb der eigenen Organisation, sowie auch innerhalb weiterer zuständigen Organisationen verbreitet werden können. Durch die schnelle Durchführung der einzelnen Schritte von der Sichtung der Situation, zur Weiterleitung an die Organisationen, können dadurch sofort weitere Maßnahmen gesetzt werden und dadurch unter Umständen Tragödien verhindert werden.

Im Rahmen dieser Masterarbeit wird ein Entwicklungskonzept für eine mobile Kommunikations- und Multisensorlösung für Sicherheits- und Risikomanagement im Freiland und im Objektschutz erarbeitet, getestet und evaluiert. Ziel ist also die Entwicklung eines energie- und kommunikationsautarken multisensoralen Systems. Hauptaugenmerk soll dabei das Monitoring im Rahmen von Sicherungsaufgaben bei urbanen Plätzen und Großveranstaltungen sein. Neben dieser genannten Hauptaufgabe, kann das System auch für den Einsatz dieser angeführten Sicherheitsszenarien verwendet werden:

- Monitoring von Transitzonen und Grenzräumen
- Überwachung von Industrieanlagen und kritischer Infrastruktur
- Feldlagerschutz im Rahmen von humanitären Einsätzen des Bundesheeres



## 1.2 Problemstellung

Um diese Multisensorlösung für das Sicherheits- und Risikomanagement entwickeln zu können, sind vorher noch einige Problemstellungen zu lösen.

- **Mastgestützte Sensorkombination**  
Evaluierung einer kostenoptimierten und nutzerorientierten Sensorkombination inklusive einer Real Time Kinematic (RTK) Lösung um eine „grobe Georeferenzierung“ der Bildbereiche für die Generierung eines echtzeitnahen Lagebildes zu ermöglichen. Eine „energieautarke“ Lösung soll für ein mobiles Mastsystem konzipiert und entwickelt werden.
- **Datenprozessierung**  
Die Bilddatenbereiche werden auf Basis der Positions- und der Orientierungsparameter aus dem RTK-System georeferenziert. Dazu ist eine entsprechende Analyse von kostengünstigen RTK-Systemen erforderlich sowie Genauigkeitsuntersuchungen für die einzelnen Einsatzszenarien.
- **Kommunikationstechnologien**  
Integration einer stabilen Kommunikationslösung (WLAN, LTE) für eine echtzeitnahe Datenübertragung, sowohl für eine lokale Datenverteilung (mobile Lösung für Einsatzkräfte Vorort), als auch eine gezielte Weiterleitung von Informationen in eine Einsatzzentrale.

## 1.3 Rahmenbedingung

Um einen flexiblen Einsatz zu gewährleisten, baut das Konzept auf innovative, energieautark betreibbare mobile Teleskopmastsysteme auf, die mit Kommunikations-, sowie verschiedenen Sensormodulen bestückt werden.

## 1 Einleitung

Der Fokus liegt auf der Demonstration der Einbindung mehrerer Netzwerke zur Kommunikation zwischen der Einsatzzentrale und einem Abschnittskommandanten, der mit einem einsatztauglichen Tablet ausgestattet ist. Dabei soll die Übermittlung von Multimediainformationen im Rahmen eines typischen Einsatzszenarios getestet werden, um Feedback zu Nutzerschnittstellen zu erhalten und weitere Anforderungen zu erheben.

In weiterer Folge wird ein kamerabasiertes System zur Verbesserung des lokalen Lagebildes entwickelt.

Das System besteht unter anderem aus einem Smartphone, welches an einer Teleskopstange montiert wird. Dadurch ermöglicht das System die Aufnahme von Bildern und Videos aus einer erhöhten Position. Zusätzlich erhält man durch das montierte Smartphone Geo- und Lagesensoren, damit sowohl der Standort, als auch die Blickrichtung der Aufnahme ermittelt werden kann. Dadurch ist es möglich die Aufnahmen innerhalb einer Karte zu verorten.

Ziel der gegenständlichen Arbeit fokussiert sich deshalb auf die Entwicklung von innovativen, technischen Lösungen für die Beobachtung von kritischen Bereichen.

### 1.4 Forschungsfragen

Die grundlegende Forschungsfrage in dieser Arbeit ist, den Unterschied bei Genauigkeitsuntersuchungen zwischen professionellen Global Positioning System (GPS) Sensoren und Smartphone GPS Sensoren, sowie einem RTK System zu ermitteln.

Dabei sollen einerseits die Positionsdaten von einem professionellen GPS Sensor, mit den Daten von verschiedenen Smartphone GPS Sensoren und einem RTK Sensor verglichen werden und dabei untersucht werden, ob die Differenz bei den Genauigkeitsmessungen bei Millimetern, Zentimetern oder unter Umständen sogar Metern liegen?

## 1 Einleitung

Neben den Genauigkeitsuntersuchungen von GPS Sensoren, soll zusätzlich noch die Genauigkeit von einem professionellen Inertial Measurement Unit (IMU) Sensor, zu verschiedenen Smartphone IMU Sensoren, sowie der RTK IMU getestet werden und dabei ermittelt werden, wie genau diese sind, beziehungsweise wie stabil sich diese verhalten?

## 2 Grundlagen

### 2.1 Smartphone

Laut der Statistik Seite „statista“ benutzten im Jahre 2017 2.32 Milliarden Menschen weltweit ein Smartphone. Diese Zahl sollte bis 2020 auf bis zu 2.87 Milliarden steigen, wie es in der Abbildung 2.1 zu sehen ist. <sup>1</sup>

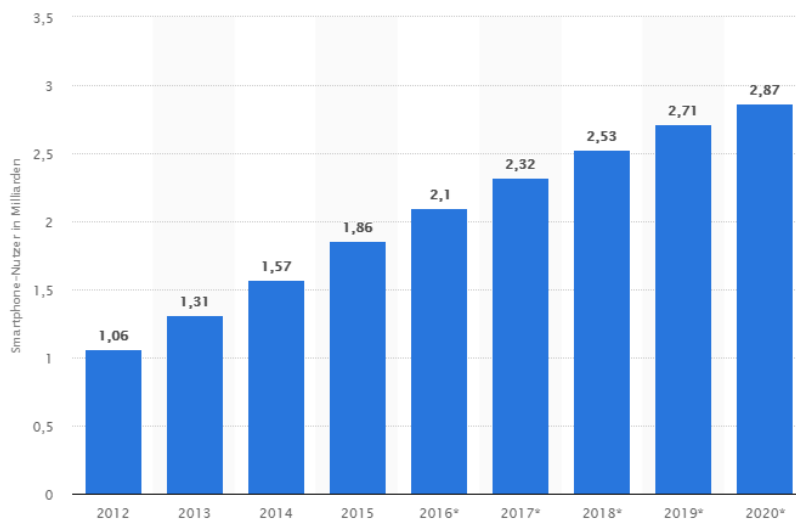


Abbildung 2.1: Die Anzahl der Smartphone Nutzer liegt im Jahre 2017 bei 2.32 Milliarden und soll bis 2020 auf bis zu 2.87 Milliarden Nutzer steigen <sup>1</sup>.

<sup>1</sup><https://de.statista.com/statistik/daten/studie/309656/umfrage/prognose-zur-anzahl-der-smartphone-nutzer-weltweit/>

## 2 Grundlagen

Ein großer Teil davon verwendet auch täglich, ob beruflich oder privat, „Apps“. Momentan befinden sich im „Play Store“ von Google rund 3.36 Millionen „Apps“, die täglich mehr werden <sup>2</sup>.

Die Anzahl der App Downloads weltweit beläuft sich im Jahr 2017 auf 197 Milliarden und soll laut einer Prognose von „statista“ im Jahr 2021 auf 352.9 Milliarden Downloads steigen, wie es in der Abbildung 2.2 zu sehen ist <sup>3</sup>.

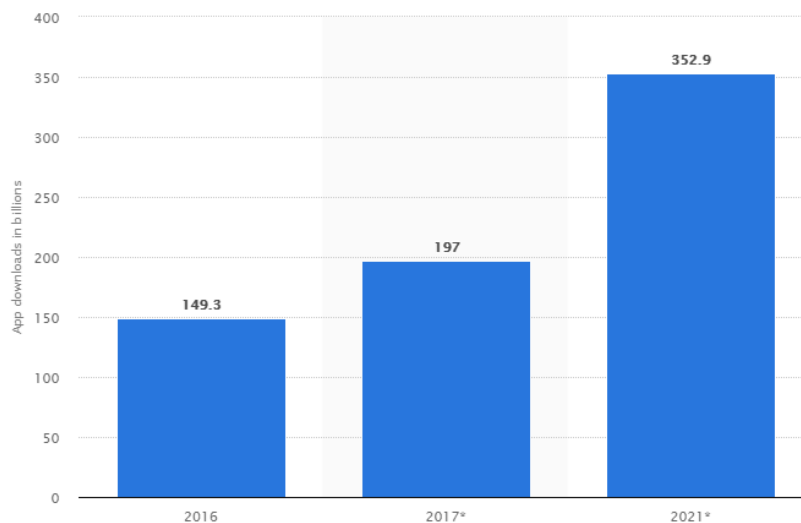


Abbildung 2.2: Die Anzahl der App Downloads weltweit beläuft sich im Jahr 2017 auf 197 Milliarden und soll im Jahr 2021 auf 352.9 Milliarden Downloads steigen <sup>3</sup>.

Aus diesem Grund ist auch die Entwicklung von „Apps“ für das Smartphone extrem wichtig, da auch ein zusätzlicher Anstieg an Smartphones und vor allem App Downloads vorhersehbar ist.

---

<sup>2</sup><https://de.statista.com/statistik/daten/studie/208599/umfrage/anzahl-der-apps-in-den-top-app-stores/>

<sup>3</sup><https://www.statista.com/statistics/271644/worldwide-free-and-paid-mobile-app-store-downloads/>

## 2.2 Smartphone Betriebssysteme

Damit ein Smartphone auch funktionieren kann, benötigt dieses ein Betriebssystem.

Dabei gibt es momentan zwei große Anbieter. Einmal Apple mit dem iOS Betriebssystem und Google mit Android.

Der Marktanteil von Android liegt momentan bei 85.1%, bei iOS liegt dieser bei 14.7% und laut Prognosen sollten diese auch, wie in der Abbildung 2.3 erkennbar, bis 2021 nahezu ident bleiben <sup>4</sup>.

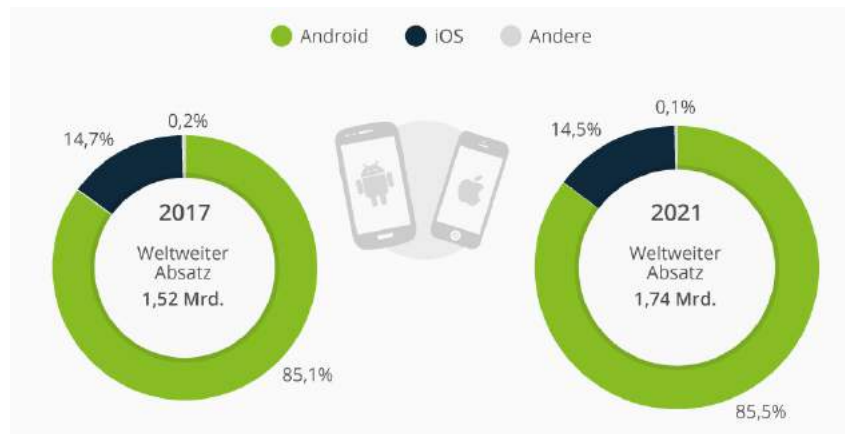


Abbildung 2.3: Der Marktanteil von Android liegt momentan bei 85.1 %, bei iOS liegt dieser bei 14.7 % und dieser soll bis 2021 nahezu ident bleiben <sup>4</sup>.

Da ein Teil dieser Arbeit auch ausschließlich für Android Smartphones entwickelt wurde, wird sich dieses Kapitel vor allem näher mit dem Android Betriebssystem beschäftigen.

<sup>4</sup><https://de.statista.com/infografik/9651/marktanteil-smartphone-betriebssysteme/>

## 2 Grundlagen

Tabelle 2.1: Android Versionen <sup>5</sup>

Versionsnamen	Versionsnummer	API-Level	Veröffentlichung
„Cupcake“	1.5	3	April 2009
„Donut“	1.6	4	September 2009
„Eclair“	2.1	7	Januar 2010
„Froyo“	2.2	8	Mai 2010
„Gingerbread“	2.3	9	Dezember 2010
„Honeycomb“	3.0	11	Februar 2011
„Ice Cream Sandwich“	4.0	14	Oktober 2011
„Jelly Bean“	4.1	16	Juni 2012
„KitKat“	4.4	19	Oktober 2013
„Lollipop“	5.0	21	November 2014
„Marshmallow“	6.0	23	Oktober 2015
„Nougat“	7.0	24	August 2016
„Oreo“	8.0	26	August 2017

### 2.2.1 Android

Android ist ein Betriebssystem für unter Anderem Smartphones, Smartwatches, Fernseher, sowie Tablets, welches von Google entwickelt wird.

Die erste Version wurde 2008 veröffentlicht, ab der Version 1.5 aus dem Jahre 2009 wurde für jede Version neben der Versionsnummer, der Name einer Süßigkeit angehängt. Dabei ist zu beachten, dass der Anfangsbuchstabe dieser Süßigkeit alphabetisch, aufsteigend ist.

Dies war im ersten Jahr „Cupcake“, danach „Donut“, und so weiter bis zur aktuellsten Version aus dem Jahre 2017 mit dem Namen „Oreo“. Eine vollständige Liste aller Versionen gibt es in der Tabelle 2.1 <sup>5</sup>.

Bei der Entwicklung von Applications ist es wichtig auf das API-Level zu schauen, denn diese bestimmt ab welcher Android Version man die App nutzen kann.

---

<sup>5</sup><https://www.android.com/intl/en.us/history/>

## 2 Grundlagen

Da viele Entwickler die Anwendung auch für Smartphones mit älteren Androidversionen zur Verfügung stellen möchten, sollte man sich vor dem Start der Entwicklung die Verteilung der verschiedenen Androidversionen, wie hier in der Abbildung 2.4, ansehen <sup>6</sup>.

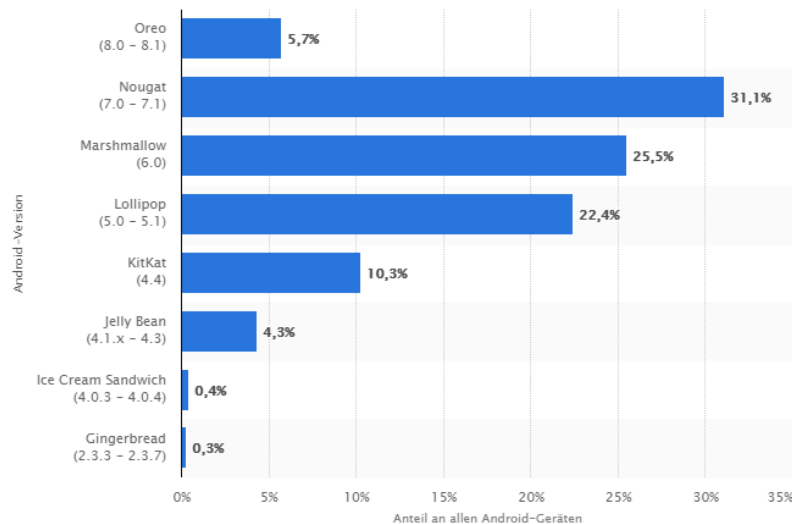


Abbildung 2.4: Die Graphik vom Mai 2018 zeigt, dass nur ein sehr kleiner Teil von 0.3%, sowie 0.4% die Versionen „Gingerbread“ und „Ice Cream Sandwich“ verwenden. Auch die Anzahl der Nutzer mit der aktuellsten Version „Oreo“ fällt mit 5.7% sehr gering aus. Die größten Anteile besitzen im Mai 2018 die Versionen „Lollipop“, „Marshmallow“ und „Nougat“, die zwischen 22.4% und 31.1% liegen <sup>6</sup>.

Die Abbildung 2.4 vom Mai 2018 zeigt, dass nur ein sehr kleiner Teil von 0.3%, sowie 0.4% die Versionen „Gingerbread“ und „Ice Cream Sandwich“ verwenden. Auch die Anzahl der Nutzer mit der aktuellsten Version „Oreo“ fällt mit 5.7% sehr gering aus.

Die größten Anteile besitzen im Mai 2018 die Versionen „Lollipop“, „Marshmallow“ und „Nougat“, die zwischen 22.4% und 31.1% liegen <sup>6</sup>.

<sup>6</sup><https://de.statista.com/statistik/daten/studie/180113/umfrage/anteil-der-verschiedenen-android-versionen-auf-geraeten-mit-android-os/>



## 2 Grundlagen

Nachdem die Verteilung analysiert wurde, muss ein Entwickler sich entscheiden, ab welcher Version die Anwendung unterstützt werden sollte.

Dabei muss unter Umständen entschieden werden, ob ich lieber die neusten Funktionen der neuen Versionen verwenden möchte, dabei aber nur einen kleinen Teil der Nutzer meine Application zur Verfügung stellen kann oder auf neue Funktionen verzichten kann und dafür einen großen Anteil aller Androidnutzer, die Anwendung bereitstellen kann [Gar11].

### 2.2.2 Architektur

Das Betriebssystem Android ist eine frei zugängliche, quelloffene Software, dies bedeutet dass deren Quelltext von jedem eingesehen, geändert und auch genutzt werden kann. Die Open Source Plattform wird zwar von Google entwickelt, gehört aber der Open Handset Alliance. Die Open Handset Alliance ist eine Gruppe aus 84 Unternehmen, welches 2007 von Google gegründet wurde und sich als Ziel die Schaffung von offenen Standards für Mobilgeräte gesetzt haben.<sup>7</sup>

Basis des Betriebssystems ist der Linux Kernel, darüber liegt der Hardware Abstraction Layer, welcher für Hardwarebausteine wie Kamera, Bluetooth und andere Sensoren zuständig ist. Neben der Android Runtime und verschiedenen Bibliotheken, besteht die Architektur auch aus einem Java API Framework, welches Methoden für die Appentwicklung bereitstellt. [Gar11]

Die grundlegende Architektur des Android Betriebssystems, kann in dieser Abbildung 2.5 eingesehen werden<sup>8</sup>.

---

<sup>7</sup><https://www.openhandsetalliance.com/index.html>

<sup>8</sup><https://developer.android.com/guide/platform/index.html>

### 2.2.3 Android Java

Für die Entwicklung von Anwendungen für Android wird die Programmiersprache Java verwendet.

Dazu gibt es eine Android SDK (Software Development Kit), die nicht die vollständige Java SE- Standardbibliothek enthält, jedoch eine gute Grundlage bildet um Androidapps entwickeln zu können.

## 2.3 Einplatinencomputer

Ein Einplatinencomputer ist ein Hardwaresystem, bei dem sich alle Komponenten, wie unter anderem der Prozessor, der Arbeitsspeicher oder auch der Festspeicher, direkt auf einer einzigen Platine befinden.

Der Vorteil von diesen Geräten ist, dass sie sehr platzsparend sind und für viele Bereiche, sowohl in der Industrie, als auch privat, sehr gute Einsatzbereiche finden.

## 2.4 Omnidirektionale Kamera

Eine Omnidirektionale Kamera, oder auch VR- bzw. 360-Gradkamera genannt, sind in der Lage Bilder von einem bestimmten Bereich so abzubilden, dass dieser aus den kompletten 360 Grad in der Horizontale besteht.

Vor allem durch die Erfindung der Virtual-Reality Brillen bekommen auch diese omnidirektionalen Kameras einen großen Aufschwung.

## 2.5 Handheld Gimbal

Ein Gimbal ist im Grunde genommen eine kardanische Aufhängung, mit dem Unterschied, dass ein Gimbal nicht nur mit dem Schwerpunkt auskommt um Objekte stabilisieren zu können, sondern zusätzlich einen Elektromotor beinhaltet, der zusammen mit der Inertial Measurement Unit (IMU), Kameras oder Smartphones ausgleicht.

Durch die eingebaute Inertial Measurement Unit (IMU), erkennt das System jegliche minimale Lageänderung und kann daraufhin den entsprechenden Elektromotor ansteuern, um das Gerät wieder in die waagrechte Position zu bringen.

Handheld Gimbals werden also vor allem dafür benötigt um auch in der Bewegung Videos aufnehmen zu können und dabei ohne den störenden, ruckartigen Bewegungen im Video auskommen.

## 2.6 Webserver

Ein Webserver stellt Webinhalte online, die dann durch eingeben der Internetadresse des Webserver mit einem Browser aufgerufen werden können.

Nach der Eingabe der entsprechenden Internetadresse im Browser, sendet dieser eine Anfrage an den Webserver. Der Webserver sieht sich die Anfrage an und sendet als Antwort beispielsweise eine HTML Seite zurück.

Im Gegensatz zu Apps, können Webseiten geräteunabhängig aufgerufen werden.

### 2.6.1 HTML

Die Hypertext Markup Language (HTML) ist eine textbasierte Auszeichnungssprache und wird dafür verwendet um Webseiten eine Struktur zu geben, damit Webbrowser diese auch interpretieren und anzeigen können.

### 2.6.2 CSS

Das Cascading Style Sheets (CSS) ist eine Stylesheet-Sprache und wurde entwickelt um den Inhalt der HTML Dateien, sowie das Design in den CSS Dateien zu trennen.

Cascading Style Sheets werden verwendet um den HTML Dokumenten ihr Layout und Stil zu geben.

### 2.6.3 JavaScript

JavaScript ist eine Skriptsprache, die vor allem für Webseiten verwendet werden. Da oftmals nicht nur die Anzeige von Inhalten auf einer Webseite benötigt wird, sondern auch mal verschiedenste Benutzerinteraktionen oder auch sonstige dynamischen Veränderung, kann man hierfür die Skriptsprache JavaScript verwenden.

## 2.7 C#

C# ist eine objektorientierte Programmiersprache, welche von Microsoft entwickelt wurde.

## 2.8 Python

Python ist eine Programmiersprache, die mit dem Ziel entwickelt wurde, die Übersichtlichkeit, sowie Einfachheit zu erhöhen.

Dies kann in diesem Beispiel gut vermittelt werden:

**C:**

```
int fakultaet(int x)
{
    if (x > 1)
        return x * fakultaet(x - 1);
    else
        return 1;
}
```

**Python:**

```
def fakultaet(x):
    if x > 1:
        return x * fakultaet(x - 1)
    else:
        return 1
```

## 2.9 Positionsbestimmungsverfahren

Um die Position eines bestimmten Objektes herauszufinden, gibt es mehrere Möglichkeiten.

Zwei davon sind per Global Navigation Satellite System (GNSS) und per Real Time Kinematic Verfahren (RTK).

### 2.9.1 GNSS

Das GNSS ist das globale Navigationssatellitensystem, welches für Positionsbestimmung verwendet wird und ein Sammelbegriff für mehrere globale Satellitensysteme ist.

- **GPS**

Das Global Positioning System (GPS) ist ein globales Satellitensystem zur Positionsbestimmung und wurde vom US-Verteidigungsministerium entwickelt. Die Planung, sowie Entwicklung startete 1973 und 1996 wurde GPS vollständig in Betrieb genommen. 24 GPS Satelliten, die per codierten Radiosignalen ihre aktuelle Position beziehungsweise ihre genaue Uhrzeit aussenden, umkreisen in 12 Stunden einmal die Erde, wovon vier davon sich jeweils auf einer der sechs Bahnebenen befinden. Diese vier Satelliten reichen vollkommen aus um eine Genauigkeit von ungefähr zehn Metern zu erhalten. [Geio5]

- **GLONASS**

GLONASS oder auch Global Navigation Satellite System wird vom Verteidigungsministerium der Russischen Föderation betrieben [HLWo8].

- **Galileo**

Galileo ist ein europäisches globales Navigationssatellitensystem, welches noch im Aufbau ist und im Jahr 2018 sollten die letzten der 30 vorgesehenen Satelliten in die Umlaufbahn gebracht werden [HLWo8].

- **Beidou**

Auch das chinesische Beidou Satellitensystem ist noch im Aufbau und sollte bis 2020 alle 35 Satelliten in die Umlaufbahn gebracht haben [HLWo8].

### 2.9.2 RTK

Um die Genauigkeit der Positionsbestimmung von einigen Metern, wie bei dem Global Positioning System, auf wenige Zentimeter reduzieren zu können, wurde die Real Time Kinematic Positionsbestimmung entwickelt.

Dieses System erweitert das globale Navigationssatellitensystem (GNSS) zusätzlich mit einer Referenzstationen, die Echtzeitkorrekturen liefert. Dadurch ist es möglich die Genauigkeit der Positionsbestimmung auf wenige Zentimeter zu reduzieren. Das GPS System funktioniert grundsätzlich so, dass die Zeit gemessen wird, bis ein Signal von einem Satelliten zum Empfänger gelangt.

Da man die genauen Umlaufbahnen der Raumfahrzeuge kennt, kann dadurch die Position auf der Erde bestimmt werden.

Aufgrund von verschiedenen Fehlerquellen, zum Beispiel das Wetter, da bei bewölktem Himmel die Signale abgebremst werden, kommt bei GPS Systemen nur eine Genauigkeit von zwei bis vier Metern zustande. [SK13] Um die Genauigkeit auf wenige Zentimeter reduzieren zu können gibt es die Real-Time Kinematic (RTK) Technologie, bei der zwei Empfänger zur Anwendung kommen.

Einmal die Basisstation, die stationär verwendet wird, sowie der Rover, der mobil verwendet wird (Abbildung 2.6).

Da die Basisstation die Fehler misst und die Korrekturen an den Rover sendet, kann mit diesem System eine Genauigkeit von wenigen Zentimetern erreicht werden <sup>9</sup>.

Aufgrund der Tatsache dass die Genauigkeit von 3 bis 5 Metern von GPS Satelliten, durch die RTK Technologie auf wenige Zentimeter verringert werden konnte, eröffnet dies auch eine Vielzahl an verschiedenen Anwendungsgebiete.

Beispielsweise können mit RTK Systemen Katastvermessungen durchgeführt werden, welche vor allem in Entwicklungsländern großen Anklang findet,

---

<sup>9</sup><https://docs.emlid.com/reach/common/tutorials/rtk-introduction/>

<sup>10</sup><http://www.anatumfieldsolutions.com/assets/images/Blog/RTK-Base-Rover-1024x571.jpg>

## 2 Grundlagen

da diese Länder teilweise nicht im Besitz von den notwendigen Ausrüstungen sind, um eine genaue Vermessung realisieren zu können [Pio13].

Aber auch in der Landwirtschaft sind RTK Systeme schon angekommen. Viele Landwirte nutzen RTK Lenksysteme in ihren Traktoren, um diverse Lebensmittel mit einer hohen Genauigkeit anbauen zu können. Dies entlastet nicht nur die Mitarbeiter, sondern bringt auch durch die effektivere Anpflanzung finanzielle Vorteile, vor allem durch die sinkenden Preise von low-cost RTK Systemen. [Wen17]

### 2.9.3 Inertial Measurement Unit (IMU)

Die IMU besteht aus mehreren Inertialsensoren und werden unter Anderem zum manövrieren von Luftfahrzeugen verwendet, findet jedoch auch Einsatz bei Navigationsgeräten, um bei elektronischen Störungen oder wenn kein GPS Signal verfügbar ist, wie in Tunneln, auszuweichen. Aber auch in Smartphones werden IMU Sensoren eingesetzt, die einerseits die Positionsgenauigkeit erhöhen, aber auch zusätzlich nützliche Informationen liefern, wie zum Beispiel die aktuelle Lage des Smartphones.

Die Inertiale Messeinheit beinhalten Beschleunigungssensoren, einem Gyroskop und manchmal auch einen Magnetometer (Abbildung 2.7). Der Beschleunigungssensor bestimmt dabei die Trägheitskraft, die auf eine Testmasse wirkt und misst dabei seine Beschleunigung. Der Gyroskop ist ein Kreiselinstrument, die dafür zuständig ist, die Ausrichtung zu bestimmen. Der Magnetometer kann durch die Messung der Stärke des Magnetfeldes der Erde zusammen mit dem Beschleunigungssensor als digitaler Kompass dienen. [Ahm+13]



## 2 Grundlagen

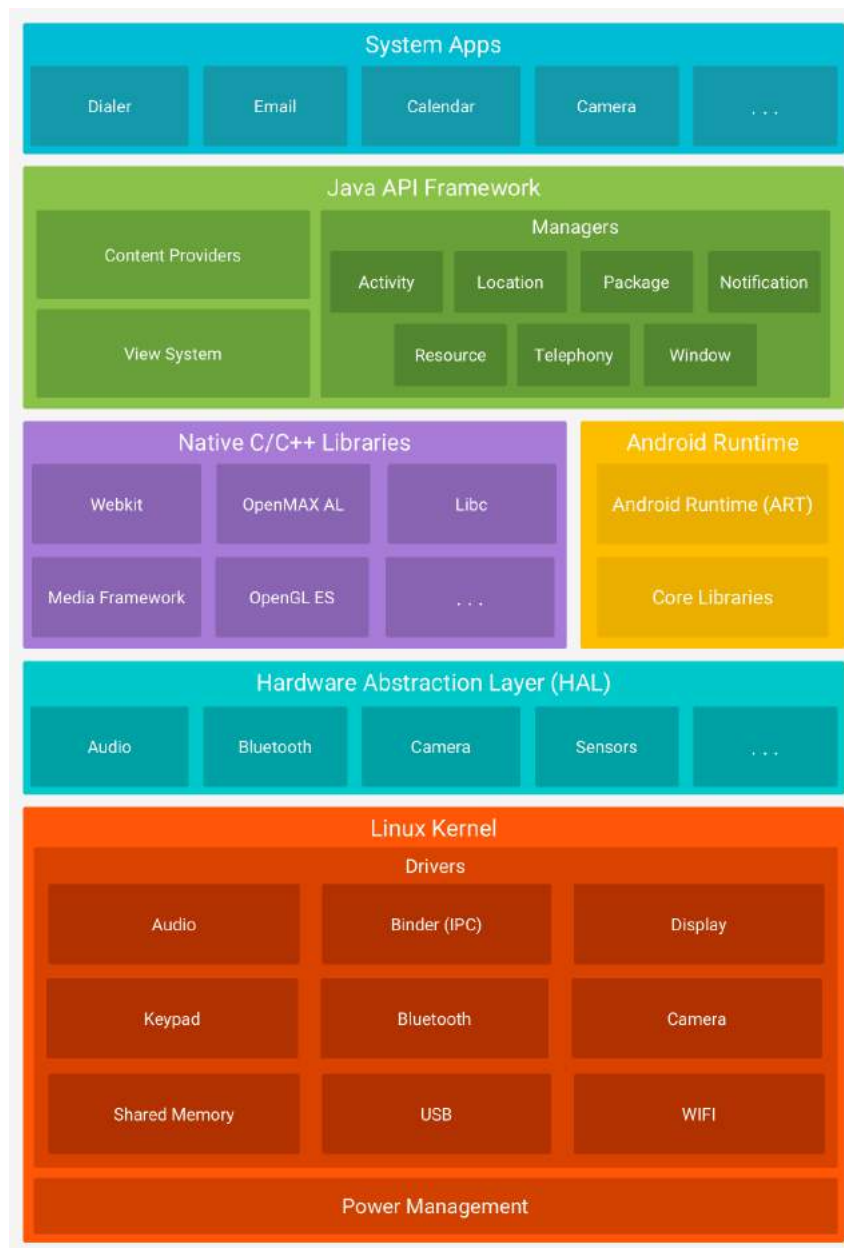


Abbildung 2.5: Basis des Betriebssystems ist der Linux Kernel, darüber liegt der Hardware Abstraction Layer, welcher für Hardwarebausteine wie Kamera, Bluetooth und andere Sensoren zuständig ist. Neben der Android Runtime und verschiedenen Bibliotheken, besteht die Architektur auch aus einem Java API Framework, welches Methoden für die Appentwicklung bereitstellt <sup>8</sup>.

## 2 Grundlagen

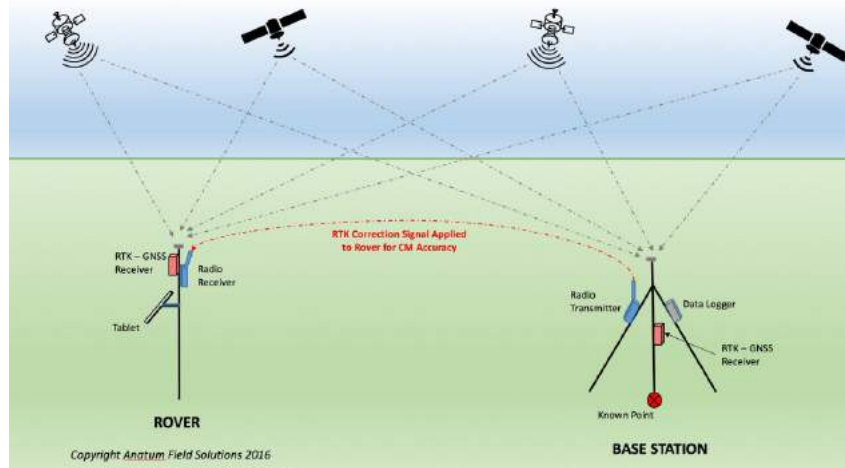


Abbildung 2.6: Um die Genauigkeit auf wenige Zentimeter zu reduzieren gibt es die Real-Time Kinematic (RTK) Technologie, bei der zwei Empfänger zur Anwendung kommen. Einmal die Basisstation, die stationär verwendet wird, sowie der Rover, der mobil verwendet wird. Da die Basisstation die Fehler misst und die Korrekturen an den Rover sendet, kann mit diesem System eine Genauigkeit von wenigen Zentimetern erreicht werden.<sup>10</sup>

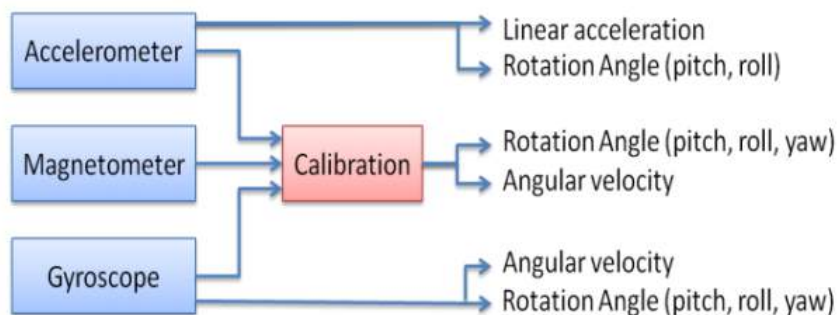


Abbildung 2.7: Die Inertiale Messeinheit beinhalten Beschleunigungssensoren, einem Gyroskop und manchmal auch einen Magnetometer. Der Beschleunigungssensor bestimmt dabei die Trägheitskraft, die auf eine Testmasse wirkt und misst dabei seine Beschleunigung. Der Gyroskop ist ein Kreiselinstrument, die dafür zuständig ist, die Ausrichtung zu bestimmen. Der Magnetometer kann durch die Messung der Stärke des Magnetfeldes der Erde zusammen mit dem Beschleunigungssensor als digitaler Kompass dienen. [Ahm+13]

## **3 Bestehende Systeme**

Das Ziel dieser Arbeit ist es eine mobile Kommunikations- und Multi-Sensorlösung für Sicherheits- und Risikomanagement im Freiland und im Objektschutz zu erarbeiten.

In diesem Kapitel wird eine Übersicht von bestehenden Systemen erstellt, die auch das Ziel haben, bei Großveranstaltungen problematische Sicherheitslagen schnellstmöglich erkennen zu können, um dementsprechend auch schnell reagieren zu können.

### **3.1 Bestehende Multi-Sensorlösungen**

Um mögliche Katastrophen bei Großveranstaltungen zu verhindern ist es wichtig diese drei Punkte zu beachten:

1. Das Problem schnell zu erkennen
2. Das Problem schnell an die zuständigen Organisationen weiterzuleiten
3. Schnellstmöglich Gegenmaßnahmen einleiten um das Problem lösen zu können

#### **3.1.1 Problemerkennung**

Zu diesem Thema gibt es schon einige Projekte, wie beispielsweise diese Arbeit, bei dem ein soziales Netzwerk verwendet wird um mögliche Sicher-

### 3 Bestehende Systeme

heitsprobleme schnell erkennen zu können.

Dafür wird beispielsweise das soziale Netzwerk Weibo verwendet, welches sich vor allem in China große Beliebtheit erfreut und mit mehr als 500 Millionen Nutzern auch die größte Plattform in China, im Bereich sozialer Medien, ist.

Weibo ist ein Kurznachrichtendienst, vergleichbar mit Twitter, mit ungefähr 50 Millionen aktiven Nutzern.

Dieses Projekt versucht die Besucher einer Großveranstaltung sozusagen als „Social Sensor“ zu verwenden, indem die Teilnehmer beispielsweise bei einem Feuerausbruch, Handgemengen oder anderen Unfällen, diese Ereignisse in dem sozialen Netzwerk Weibo veröffentlichen.

Diese großen Daten werden nun analysiert und nach dem 5W Modell, also Wer, Was, Wann, Wo und Warum, ausgewertet um schnellstmöglich das genaue Problem zu erkennen. [Zhe+16]

Ein ähnliches Projekt gibt es mit dem Kurznachrichtendienst Twitter, denn auch hier werden die sogenannten „tweets“ der Menschen ausgewertet, um mögliche Katastrophen schnell erkennen zu können [Vie10].

Ein weiteres Projekt um die Sicherheit bei Großveranstaltungen zu erhöhen ist das Projekt AGETOR. Bei diesem Konzept wird versucht den Einsatzkräften vor Ort zu unterstützen, indem denen Informationen über Personenbewegungen mitgeteilt werden.

Die Personenbewegungen werden durch die Verwendung von Mobilfunkdaten in Echtzeit ausgewertet und dadurch können die Bewegungsströme erkannt werden. Mit Hilfe dieser Informationen können die Einsatzkräfte bei Sicherheitsgefahren rechtzeitig eingreifen und eine mögliche Katastrophe verhindern. [Cik+16]

#### 3.1.2 Problemweiterleitung bzw. Problemlösung

Nachdem das Problem erkannt wurde, gibt es die nächste Schwierigkeit zu lösen, nämlich die Problemweiterleitung.

Bei Großveranstaltungen gibt es eine Vielzahl an Organisationen, die für die

### 3 Bestehende Systeme

Sicherheit und Gesundheit der Besucher zur Verfügung stehen, angefangen vom Sicherheitspersonal, Polizei, Rettung, sowie weiteren Hilfsorganisationen.

Damit auch alle zuständigen Organe, auch schnellstmöglich von der möglichen Sicherheitslage erfahren, wird ein System benötigt mit dem rasch alle Organisationen von der möglichen Problematik erfahren.

Ein Beispiel für solch ein System IT-System, ist jenes von dem Fraunhofer Institut für Arbeitswirtschaft und Organisation (IAO).

Diese beschäftigen sich mit einem Portal, welches einen rollenbasierten Informationsaustausch zwischen den beteiligten Organisation ermöglicht und diese bei der Umsetzung der einzelnen Prozesse unterstützt. [FER10]

Neben dem Informationsaustausch zwischen den einzelnen Hilfsorganisationen, ist es auch wichtig den Besuchern bei größeren auftretenden Problemen, wie Massenpaniken, Unwetter oder ähnlichem, Informationen zuzusenden, damit diese auch wissen was zu tun ist und vor allem keine unnötigen Panikreaktionen auftreten, wie es leider in der Vergangenheit schon oft der Fall war.

Daher wird in diesem Projekt versucht, den Besuchern bei Notfallsituationen über ein einheitliches System Informationen über den Ablauf und dem Geschehen zu informieren. [Roß+08]

## 3.2 Genauigkeitsmessverfahren

### 3.2.1 GPS

Ein Beispiel eines GPS Genauigkeitstests stammt aus einer Arbeit, bei der das Samsung Galaxy S, das Motorola Droid X, sowie das iPhone 4 verwendet wurden.

Dabei wurde in Chicago, Illinois eine Strecke von 1.58km, zu verschiedenen Tageszeiten abgefahren.

Dabei kann man in der Abbildung 3.1 erkennen, dass 95% der Zeit, die Positionsgenauigkeiten der Smartphones innerhalb von 10 Metern lagen.

### 3 Bestehende Systeme

[Men+11]

Beim zweiten Beispiel kann erkannt werden, dass nicht nur die Hardware, sondern auch die Software für die Positionsgenauigkeit des Smartphones wichtig ist.

In dieser Arbeit wurden neun verschiedene Fitnessapps, unter anderem Runtastic, Runkeeper, Sports Tracker oder auch MyTracks verwendet und die GPS Genauigkeit miteinander verglichen.

Als Smartphone wurde das HTC Desire Bravo verwendet.

Dabei wurde jedes Mal ein exakt gleicher Weg im Ausmaß von einem Kilometer gelaufen und am Ende wurde die Distanz der einzelnen Anwendungen verglichen.

Das Ergebnis, wie in der Abbildung 3.2 sichtbar, zeigt Unterschiede zwischen 0 und 60 Metern, obwohl das selbe Smartphone mit den gleichen Sensoren verwendet wurden. [Bau13]

Hier ein weiteres Beispiel, bei denen die Genauigkeit von Smartphone GPS Sensoren getestet wurde. Dabei wurden vier verschiedene Android Geräte verwendet, nämlich das Samsung Galaxy S2 Plus, Samsung Galaxy S3 Mini, Samsung Galaxy Tab 1 und das Google Nexus Tablet.

Es wurden verschiedene Punkte in der realen Welt ausgewählt, die Geräte wurden dort aufgestellt um dort die genaue Position zu erhalten und daraufhin wurde der Unterschied zwischen der realen Position und der gemessenen GPS Position verglichen.

Dabei gab es Positionsgenauigkeiten die zwischen 2 Metern und 8 Metern lagen (Abbildung 3.3). Dabei schnitt das Samsung Galaxy Tab mit einer durchschnittlichen Positionsgenauigkeit von 2.62 Metern am besten ab, währenddessen das Google Nexus mit einem Genauigkeitsunterschied von durchschnittlich 7.90 Metern am schlechtesten abschnitt. [Ven14]

#### 3.2.2 IMU

Ein Beispiel eines IMU Genauigkeitstests stammt aus einer Arbeit, bei der das Samsung Galaxy S2 Plus, Samsung Galaxy S3 Mini, Samsung Galaxy

### 3 Bestehende Systeme

Tab 1 und das Google Nexus Tablet verwendet wurden. Dabei wurden die Daten von dem einfachen Kompass, des Magnetometer/Accelerometer Kompasses und der Sensor Fusion von den verschiedenen Geräten aufgenommen und ausgewertet.

Dabei wurde festgestellt, dass die durchschnittliche Genauigkeit beim einfachen Kompass zwischen 8 und 14 Grad lag, bei dem Magnetometer/Accelerometer Kompass zwischen 8 und 42 Grad und die Sensor Fusion hatte eine Genauigkeit zwischen 7 und 13 Grad (Abbildung 3.4).

Am besten hat bei diesem Test das Samsung Galaxy Tab abgeschlossen mit einer Genauigkeit, die je nach Testmethode, zwischen 6.71 und 8.25 Grad lag. [Ven14]

Ein weiterer IMU Test wurde durchgeführt, indem die Genauigkeit von dem IMU Sensor alleine und in der Kombination von IMU, Barometer und Kompass getestet wurde.

Nun wurde ein bestimmter Kurs abgegangen, die empfangenen IMU-Daten aufgenommen und diese danach ausgewertet.

Dabei wurde festgestellt, dass die Sensor Fusion von IMU, Barometer und Kompass stabiler und genauer ist, als nur der einzelne IMU Sensor (Abbildung 3.5). Der Unterschied von roll und pitch Werten von dem einzelnen IMU Sensor zu der Kombinationslösung waren sehr gering und lagen im Bereich von maximal 7 Grad, jedoch die Genauigkeit zwischen den heading Werten waren extrem groß. [Gre+06]

### 3 Bestehende Systeme

FIGURE 1a - CONFIDENCE RADII OF THE SAMSUNG GALAXY S GPS DATA

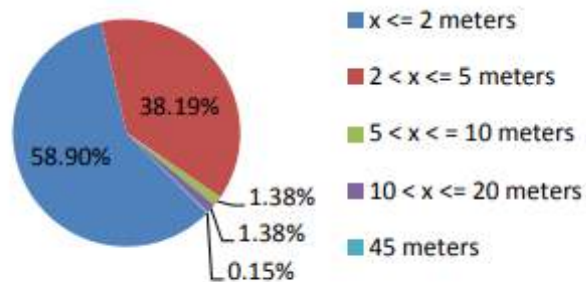


FIGURE 1b - CONFIDENCE RADII OF THE MOTOROLA DROID X GPS DATA

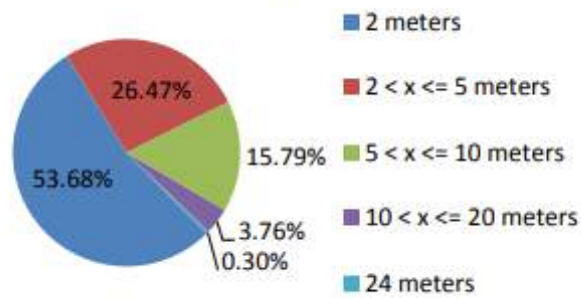


FIGURE 1c - CONFIDENCE RADII OF THE MOTOROLA DROID X GPS DATA

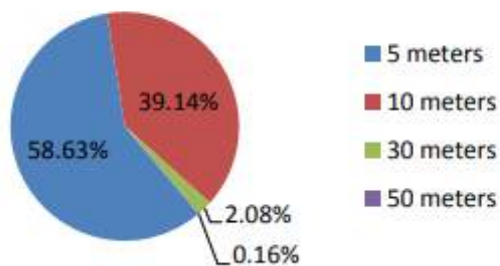


Abbildung 3.1: In dieser Abbildung kann man erkennen, dass 95% der Zeit, die Positions-genauigkeiten der Smartphones innerhalb von 10 Metern lagen [Men+11].



### 3 Bestehende Systeme

Application	Distance in meters	Deviation in meters	Rank
Adidas miCoach	1000	0	1
Endomondo	940	60	8
MapMyRun GPS Running	1030	30	6
MyTracks	1030	30	6
Noom Cardio Trainer	1010	10	2
Orux Maps	1010	10	2
Runkeeper	980	20	5
Runtastic	940	60	8
Sports Tracker	990	10	2

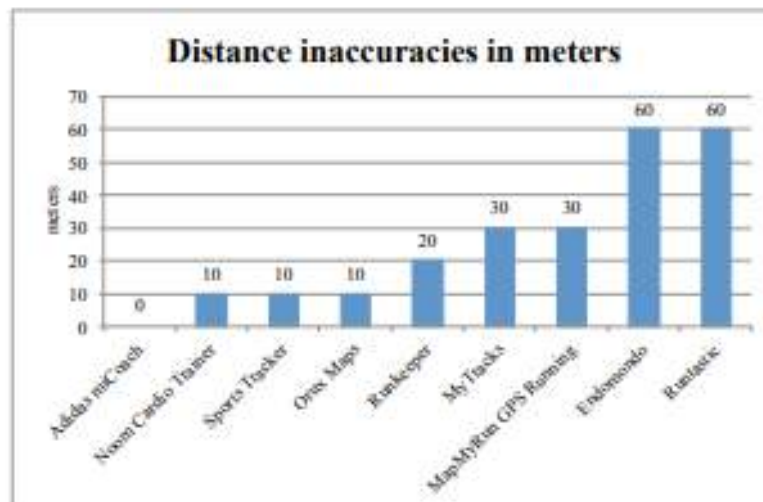


Abbildung 3.2: Das Ergebnis zeigt Unterschiede zwischen 0 und 60 Metern, obwohl das selbe Smartphone mit den gleichen Sensoren verwendet wurden [Bau13].

### 3 Bestehende Systeme

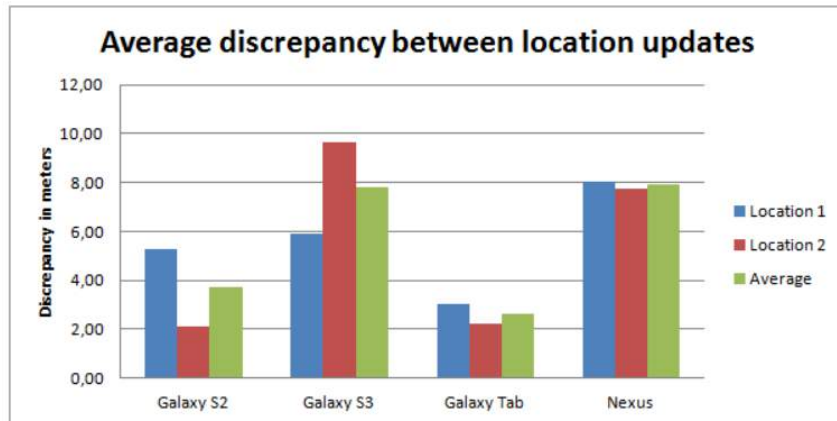


Abbildung 3.3: Wie in der Graphik ersichtlich gab es Positionsgenauigkeiten die zwischen 2 Metern und 8 Metern lagen. Dabei schnitt das Samsung Galaxy Tab mit einer durchschnittlichen Positionsgenauigkeit von 2.62 Metern am besten ab, währenddessen das Google Nexus mit einem Genauigkeitsunterschied von durchschnittlich 7.90 Metern am schlechtesten abschnitt. [Ven14]

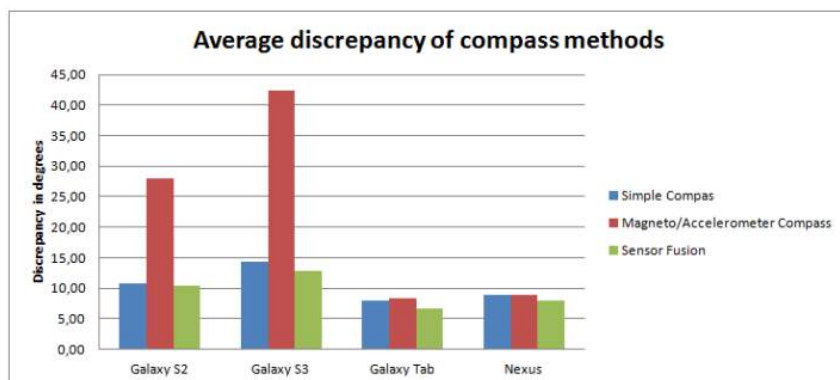


Abbildung 3.4: Wie in der Graphik ersichtlich, erkennt man dass die durchschnittliche Genauigkeit beim einfachen Kompass zwischen 8 und 14 Grad lag, bei dem Magnetometer/Accelerometer Kompass zwischen 8 und 42 Grad und die Sensor Fusion hatte eine Genauigkeit zwischen 7 und 13 Grad. Am besten hat bei diesem Test das Samsung Galaxy Tab abgeschlossen mit einer Genauigkeit, die je nach Testmethode, zwischen 6.71 und 8.25 Grad lag. [Ven14]

### 3 Bestehende Systeme

		30sec		120sec			30sec		120sec	
		Mean	Std	Mean	Std		Mean	Std	Mean	Std
IMU Only	N [m]	66.9	79.0	2472.7	3288.9	R [°]	4.1	4.6	6.6	8.2
	E [m]	334.6	456.0	6542.0	8910.6	P [°]	9.3	10.6	16.5	19.0
	U [m]	72.1	93.0	1331.4	1857.5	H [°]	125.6	143.3	184.3	215.0
IMU +Barometer +Compass	N [m]	476.7	616.9	3038.4	3731.9	R [°]	4.1	4.8	5.0	6.1
	E [m]	93.8	142.3	673.3	787.5	P [°]	6.8	8.3	10.2	12.5
	U [m]	11.3	11.9	4.8	6.3	H [°]	0.5	0.7	0.5	0.5

Abbildung 3.5: Wie in der Graphik ersichtlich wurde festgestellt, dass die Sensor Fusion von IMU, Barometer und Kompass stabiler und genauer ist, als nur der einzelne IMU Sensor. Der Unterschied von roll und pitch Werten von dem einzelnen IMU Sensor zu der Kombinationslösung waren sehr gering und lagen im Bereich von maximal 7 Grad, jedoch die Genauigkeit zwischen den heading Werten waren extrem groß. [Gre+06]

## 4 Methodik

Beschreibung	Dauer	Starttermin
Evaluierung (RTK-System, Kamera, Positionssensoren)	3 Wochen	22.05.2017
Literaturrecherche	Laufend	
Beschreibung der Problemstellung	1 Woche	12.06.2017
1. Meilenstein		
Entwicklung eines ersten Prototypen	2 Monate	19.06.2017
2. Meilenstein		
Demonstration Frequency	14.08.2017 - 17.08.2017	
Evaluierung	1 Woche	21.08.2017
Einarbeitung der Anmerkungen aus der Demonstration (Frequency)	1,5 Monate	28.08.2017
3. Meilenstein		
Entwicklung des zweiten Prototypen	1 Monat	09.10.2017
Evaluierung	1 Woche	06.11.2017
4. Meilenstein		
GPS/IMU Testvorbereitungen	3 Wochen	30.04.2018
GPS/IMU Genauigkeitstests	1 Woche	21.05.2018
Auswertung der Testergebnisse	3 Wochen	28.05.2018
5. Meilenstein		
Schreiben	3 Monate	02.07.2018

Abbildung 4.1: Zeitplan Masterarbeit

Der erste Schritt der Arbeit ist die ausführliche Evaluierung diverser Sensoren, welche dann für das fertige Mastsystem verwendet werden sollen.

Ziel ist es eine kompakte, unabhängige Hardwarelösung für das Mastsystem zu finden. Das System soll einerseits aus Kameras bestehen, die die

aktuelle Lage einerseits als Fotos, aber auch als Videos aufnehmen sollten. Gleichzeitig ist auch eine 360 Grad Kamera angedacht um sehr schnell, das komplette Gelände überwachen zu können. Da die Fotos, als auch die Videos als Livestream an die Zentrale gesendet werden sollten, wird auch zusätzlich Hardware benötigt, die diese Aufgaben erledigen sollen.

Außerdem wird eine Positionsbestimmung, als auch eine Richtungsbestimmung benötigt. Die Kamera sollte auch im fahrenden Betrieb wackelfreie Bilder und Videos liefern, daher wird auch noch eine Stabilisierungshardware benötigt, die dies ermöglichen soll.

Nach der ersten groben Recherche, welche Hardwarelösungen ungefähr für das System angedacht werden, begann die Arbeit an dem ersten Prototypen.

### 4.1 Prototyp 1

Wir bekamen die Chance die ersten Tests auf einer Großveranstaltung, nämlich dem Frequency Festival in St. Pölten, zu testen. Das Frequency Festival ist ein Musikfestival, welches jährlich seit 2001 stattfindet und 2009 von Salzburg nach St. Pölten übersiedelt ist. Im Jahr 2017 kamen 140.000 Menschen, daher war dies auch eine perfekte Möglichkeit unser Mastsystem zu testen <sup>1</sup>.

Da die ersten Tests am Frequency schon sehr knapp bevorstanden, wurde für den ersten Prototyp ein System entwickelt, welches sehr einfach, mit so wenig zusätzlicher Hardware wie möglich auskommen sollte.

#### 4.1.1 Hardwarerecherche

- Kameras

---

<sup>1</sup><https://www.frequency.at>

## 4 Methodik

Ziel ist es eine Kamera zu finden, die eine leichte Anbindung ermöglicht, mit der man leicht und schnell an die Daten kommt, beziehungsweise auch die Möglichkeit eines Livestreames anbietet.

Die ersten Versuche begannen mit einer GoPro Hero 4. Die Anbindung erwies sich als sehr einfach, da GoPro eine eigene python Library zur Verfügung stellt, mit der es möglich ist Fotos, sowie Videos aufzunehmen, die Kamera ein- beziehungsweise auszuschalten und noch einige weitere nützliche Funktionen. Leider ist es nicht möglich, mit der GoPro einen Livestream aufzubauen, daher ist diese Kamera nicht die gewünschte Hardware, die für unser System benötigt wird.

Als nächstes wurde im Internet nach weiteren externen Kameras recherchiert, die neben einer guten und leichten Anbindung, auch die Möglichkeit besitzen, einen Livestream zu erstellen. Leider erzielte diese Recherche keine optimale Lösung für unser System. Daher wurde entschieden ein Smartphone zu nehmen, denn dieses hat den Vorteil einer leichten und schnellen Anbindung und bietet außerdem die Funktionalitäten eines Livestreames an.

Nachdem eine normale Kamera gefunden wurde, wurde noch eine 360 Grad Kamera benötigt. Die Anforderungen waren die selben, denn auch hier wird eine leichte Anbindung, als auch die Möglichkeit eines Livestreams benötigt. Nach einer ausführlichen Hardwarerecherche entschieden wir uns für die Ricoh Theta R (Abbildung 4.2), die neben der Full HD Auflösung, auch einen HD Livestream anbietet.

- **Positions- und Richtungsbestimmung**

Für die Positions- und Richtungsbestimmung wurden verschiedene externe GPS, beziehungsweise IMU Sensoren getestet, da aber die Anbindung für den ersten Prototyp sich als zu aufwändig herausstellte, wurde entschieden auch hierfür auf die internen GPS, beziehungsweise IMU Sensoren des Smartphones zurückzugreifen.

Die Anbindung funktioniert sehr einfach und die Genauigkeit bei

---

<sup>2</sup><https://ricohr.ricoh/img/productImage.png>

## 4 Methodik



Abbildung 4.2: Produktbild von der Ricoh Theta R <sup>2</sup>

den Smartphonesensoren ist für den ersten Prototypen absolut ausreichend.

- **Streaminghardware**

Damit die Fotos, als auch der Videostream von den Kameras auch in die Zentrale übertragen werden können, wird noch eine Hardware benötigt, die diese Aufgaben lösen kann.

Da eine kleine, kompakte und zuverlässige Hardware für das Mastsystem benötigt wird, legte man sich nach einer Recherche auf einen Einplatinencomputer fest, wobei man sich entweder auf einen Raspberry Pi oder einem ODROID festlegte.

Dazu wurde eine Übersicht, wie hier sichtbar in der Tabelle 4.1 erstellt, um so eine genaue Gegenüberstellung der wichtigsten Hardwareteile zu erhalten.

Aufgrund des besseren Prozessors, den größeren Arbeitsspeicher, als auch der externen Kühlung, haben wir uns für den ODROID entschieden

---

<sup>2</sup><https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>

<sup>3</sup><https://www.ODROID.co.uk/ODROID-c2-cat/hardkernel-ODROID-c2-board>

<sup>4</sup><https://www.raspberrypi.org/app/uploads/2017/05/Raspberry-Pi-3-462x322.jpg>

<sup>5</sup><https://www.ODROID.co.uk/image/cache/catalog/liymo/ODROID/XU4/XU4->

## 4 Methodik

Tabelle 4.1: Raspberry Pi <sup>2</sup> vs. ODROID <sup>3</sup>

	<b>Raspberry Pi - Pi 3 Model B</b>	<b>ODROID - C2</b>
<b>CPU</b>	Quad Core 1.2GHz Broadcom BCM2837 64bit CPU	AmlogicS905 (ARM® Cortex®-A53 (ARMv8) 1.5Ghz quad core CPU)
<b>GPU</b>	Broadcom Dual Core VideoCore IV	ARM Mali™-450
<b>RAM</b>	1 GB	2 GB
<b>USB</b>	4 x USB 2.0	4 x USB 2.0



Abbildung 4.3: Raspberry Pi <sup>4</sup>

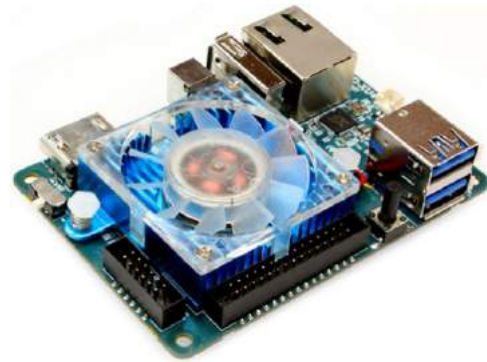


Abbildung 4.4: ODROID <sup>5</sup>

- **Stabilisierungshardware**

Wichtig ist auch eine Stabilisierungshardware, damit das Smartphone auch bei Bewegung ruhige Videos aufnehmen kann. Dafür entschied man sich auf die Osmo Mobile von dem chinesischen Technologieunternehmen DJI, bei der die Steuerung, als auch die Anbindung sehr gut anzusteuern ist.

- **Steuerungshardware**

Da wir uns schon bei der Kamera beziehungsweise bei der Positions-

---

500x500.jpg

<sup>7</sup><https://www1.djicdn.com/assets/images/products/osmo-mobile/v2/so-bg-black2-f4e828521f446cef7a91743614f7e7c3.png?from=cdnMap>



## 4 Methodik



Abbildung 4.5: Osmo Mobile 7

und Richtungsbestimmung auf das Smartphone entschieden haben, werden wir dieses auch als Steuerungshardware einsetzen. Damit es auch die Möglichkeit gibt, bei Problemen etc. direkt auf bestimmte Funktionalitäten zugreifen zu können, ist das Ziel auf dem Smartphone einen Webserver aufzusetzen, mit der jedes Endgerät, im selben Netzwerk, Zugriff auf bestimmte Funktionen bekommt.

- **Smartphone**

Im letzten Schritt wird nun noch ein Smartphone gesucht, welches einerseits eine gute Kamera besitzt, als auch gute Hardware um alle Funktionen, die dem Handy übertragen werden, standesgemäß ausgeführt werden können.

Dabei haben wir uns für das Sony Z5 entschieden, da es alle von uns gestellten Anforderungen erfüllt.

---

<sup>8</sup>[https://www.gsmarena.com/sony\\_xperia\\_z5-7534.php](https://www.gsmarena.com/sony_xperia_z5-7534.php)

<sup>9</sup><https://images-na.ssl-images-amazon.com/images/I/81TZOZHPpoL..SL1500..jpg>

## 4 Methodik

Tabelle 4.2: Hardware-Merkmale des Sony Z5 <sup>8</sup>

	<b>Sony Z5</b>
<b>CPU</b>	Qualcomm® Snapdragon™ 810
<b>GPU</b>	Adreno™ 430
<b>RAM</b>	3 GB
<b>Akku</b>	2.900 mAh
<b>Hauptkamera</b>	23 Megapixel
<b>Frontkamera</b>	5 Megapixel



Abbildung 4.6: Sony Z5 <sup>9</sup>

### 4.1.2 Softwareimplementierung

In der Abbildung 4.7 wird die Konfiguration der ausgewählten Hardware bildlich dargestellt.

Außerdem werden die Verbindungen der einzelnen Komponenten, ob per WLAN, Ethernet oder USB gekennzeichnet.

Um die verschiedenen ausgewählten Hardwarekomponenten auch zentral ansprechen zu können, wird eine Android App entwickelt, die alle Komponenten steuern soll.

## 4 Methodik

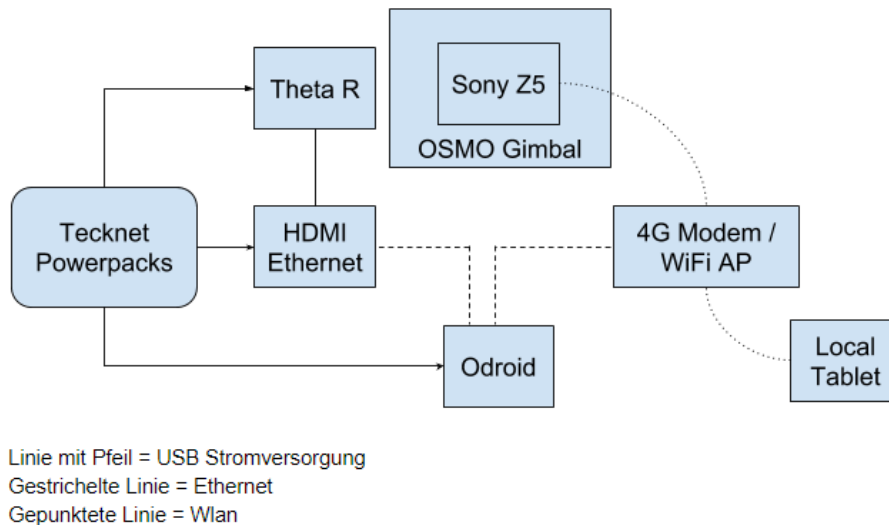


Abbildung 4.7: In dieser Abbildung wird die Konfiguration der ausgewählten Hardware bildlich dargestellt. Außerdem werden die Verbindungen der einzelnen Komponenten, ob per WLAN, Ethernet oder USB gekennzeichnet

- **Kameras**

Der erste Schritt war es die Smartphone Kamera anzusprechen. Dafür wurde eine Android Application erstellt, mit der es möglich war Fotos auszulösen, als auch abzuspeichern.

Wie man in der Abbildung 4.8 sieht, ist die Abdeckung von der Version Nougat, aus dem Jahre 2016 nur bei 1.5%. Daher wurde die App mit der Mindest API Version von 16 implementiert, um eine Abdeckung von 99.2% zu erhalten.

Der nächste Schritt lag darin, das Bild, welches das Smartphone aufnimmt, als Livestream anzubieten, damit es auch andere im selben Netzwerk ansehen können. Dafür wurden einerseits die einzelne Frames von der Kamera benötigt, diese mussten dann komprimiert und dann über ein Protokoll übertragen werden.

Die Frames konnten sehr schnell von der Kamera empfangen werden,

## 4 Methodik

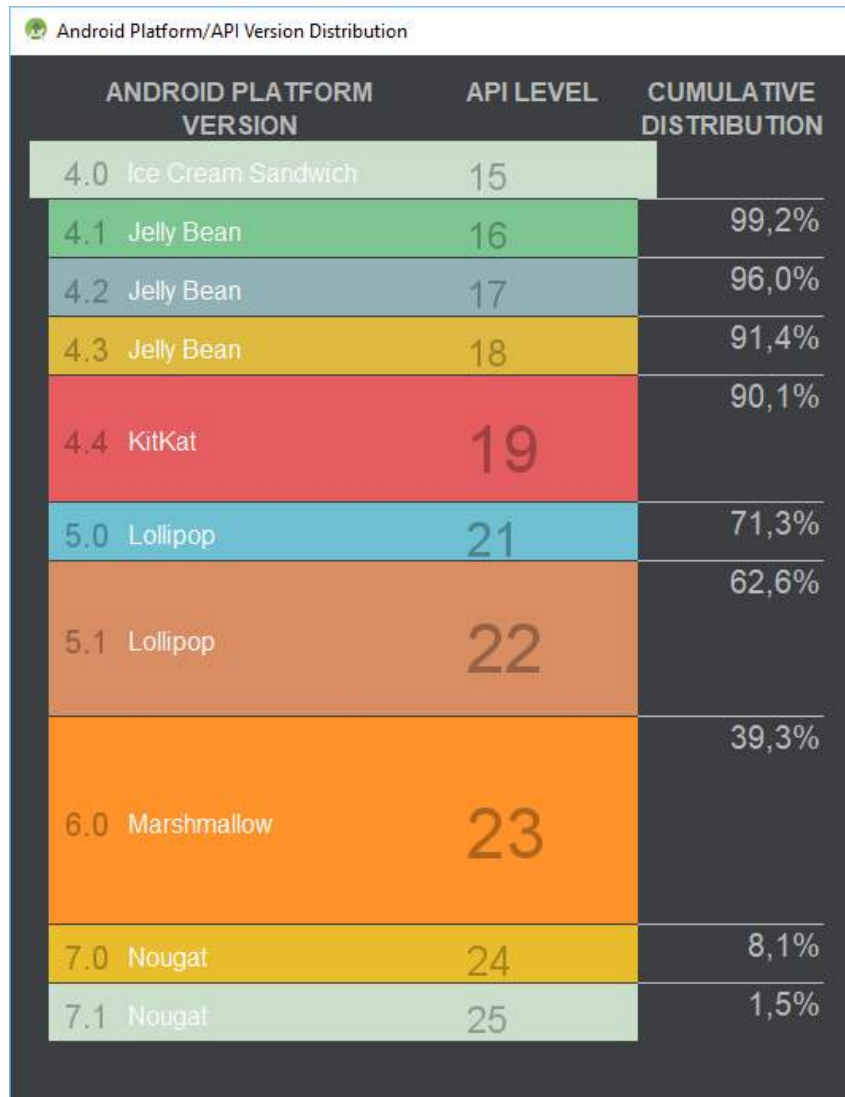


Abbildung 4.8: Wie man in der Abbildung sieht, ist die Abdeckung von der Version Nougat, aus dem Jahre 2016 nur bei 1,5%, die von der der Version 16 bei 99,2%

es gab aber noch zwei wichtige Fragen. Einerseits, welche Komprimierung ist für diese Art die Beste, als auch welches Streaming-Protokoll ist für diese Aufgabe am ehesten geeignet.

## 4 Methodik

Als Komprimierungsformat wurde Motion JPEG verwendet, aus dem Grund, da bei diesem Komprimierungsverfahren jedes Bild separat als JPEG Bild komprimiert wird und da wir neben dem Livestream auch Einzelbilder anbieten, konnten wir dadurch das komprimierte Bild einerseits für den Livestream, als auch für das Einzelbild verwenden.

Außerdem ermöglicht uns dieses Komprimierungsformat die genaue Einstellung der Qualität, um auch bei geringer Bandbreite den Livestream problemlos anbieten zu können.

Bei dem Übertragungsprotokoll entschieden wir uns auf HTTP, da das Video auch auf dem WebServer laufen sollte, und die Einbindung per HTTP sehr komfortabel für diese Aufgabe ist.

Da Ricoh neben einer SDK, auch eine Beispielsapp für Android zur Verfügung stellt, war die Anbindung der 360 Grad Kamera sehr einfach durchführbar.

- **Positions- und Richtungsbestimmung**

Die Positions- und Richtungsbestimmung mit dem Smartphone erfolgte bei dem ersten Prototypen per Android App und diese Anbindung ging auch dank der guten Dokumentation von Android sehr einfach über die Bühne.

### **Bewegungssensoren <sup>10</sup>:**

#### TYPE\_ACCELEROMETER

Beschleunigungskraft entlang der x/y/z-Achse (einschließlich der Schwerkraft). (values[0...2])

#### TYPE\_ACCELEROMETER\_UNCALIBRATED

Gemessene Beschleunigung entlang der x/y/z-Achse ohne jeglicher Kompensation. (values[0...2]) Gemessene Beschleunigung entlang der x/y/z-Achse mit geschätzter Kompensation. (values[3...5])

---

<sup>10</sup>[https://developer.android.com/guide/topics/sensors/sensors\\_motion](https://developer.android.com/guide/topics/sensors/sensors_motion)

### TYPE\_GRAVITY

Schwerkraft entlang der x/y/z-Achse. (values[0...2])

### TYPE\_GYROSCOPE

Rotationsgeschwindigkeit der x/y/z-Achse. (values[0...2])

### TYPE\_GYROSCOPE\_UNCALIBRATED

Rotationsgeschwindigkeit (ohne Driftkompensation) der x/y/z-Achse. (values[0...2]) Geschätzter Drift der x/y/z-Achse. (values[3...5])

### TYPE\_LINEAR\_ACCELERATION

Beschleunigungskraft entlang der x/y/z-Achse (ohne Schwerkraft). (values[0...2])

### **Positionssensoren <sup>11</sup>:**

#### TYPE\_ORIENTATION

Azimuth (Winkel um die z-Achse) (values[0]) Pitch (Winkel um die x-Achse) (values[1]) Roll (Winkel um die y-Achse) (values[2])

Damit konnten die GPS Daten, als auch die IMU Daten des Smartphones kontinuierlich empfangen und abgespeichert werden.

Diese Daten wurden als GeoJSON File abgespeichert, um die Weiterverarbeitung, sowie Übersichtlichkeit zu erleichtern.

Das GeoJSON Format ist ein offenes Format zum Kodieren von einer Vielzahl an geographischen Datenstrukturen.

```
{
  "type": "Feature",
  "geometry": {
    "type": "Point",
    "coordinates": [47.063872, 15.452955]
  },
  "properties": {
```

---

<sup>11</sup>[https://developer.android.com/guide/topics/sensors/sensors\\_position](https://developer.android.com/guide/topics/sensors/sensors_position)

#### 4 Methodik

```
    "name": "JOANNEUM RESEARCH – DIGITAL"  
  }  
}
```

Hier Beispieldaten, wie das GeoJSON File übermittelt wird:

```
{  
  "type": "FeatureCollection",  
  "image_resolution_h": 1280,  
  "image_resolution_v": 720,  
  "focal_length_in_pixel": 792.1,  
  "features":  
  [  
    {"type": "Feature",  
      "geometry":  
        {"type": "Point",  
          "coordinates": [15.45307624, 47.06424597, 431]}},  
      "properties":  
        {"id": 0,  
          "yaw": -49.69215,  
          "pitch": -103.52163,  
          "roll": -62.338127,  
          "time": 1501751331054}}},  
    {"type": "Feature",  
      "geometry": {"type": "Point",  
          "coordinates": [15.45307628, 47.06424598, 431]}},  
      "properties":  
        {"id": 1,  
          "yaw": -40.127453,  
          "pitch": -97.56442,  
          "roll": -63.431904,  
          "time": 1501751331608}}},  
    {"type": "Feature",  
      "geometry": {"type": "Point",  
          "coordinates": [15.45307628, 47.06424598, 431]}},  
      "properties":  
        {"id": 2,
```

```
        "yaw": -36.197105,  
        "pitch": -95.303505,  
        "roll": -63.44429,  
        "time": 1501751332408}}  
    ]  
}
```

- **Streaminghardware**

Damit auch der ODROID verwendet werden kann, muss dieser zuerst aufgesetzt werden. Als Betriebssystem entschieden wir uns für Ubuntu Mate 16.04.

Ubuntu ist eine Linux Distribution, welches auf das gemeinschaftlich entwickelte freie Betriebssystem Debian basiert.

Als erster Schritt musste das Ubuntu Mate File heruntergeladen werden <sup>12</sup>.

Danach wird das File entpackt und auf die SD Karte gespeichert und ODROID kann nun mit Ubuntu Mate gestartet werden.

Nun hat man die Möglichkeit alle weiteren benötigten Programme per Konsole mit: `sudo apt-get install „program name“` einfach zu installieren

Der nächste Schritt lag darin die Streamingkonfiguration, wie in der Abbildung 4.9 ersichtlich, zu erstellen.

Dabei wurde einerseits die Komponenten ausgewählt, die das Videomaterial aufnehmen, andererseits auch das Übertragungsprotokoll, sowie die einzelne Verteilung, wie der Livestream schlussendlich an die einzelnen Nutzer verteilt wird.

Bezüglich der Videoübertragung unterscheiden wir zwischen den zwei Varianten:

### 360 Grad Videostream

Die Theta R sendet in 1920x960 und 30 FPS die raw Videodaten zu

---

<sup>12</sup>[http://de.eu.odroid.in/ubuntu\\_16.04lts/](http://de.eu.odroid.in/ubuntu_16.04lts/)



## 4 Methodik

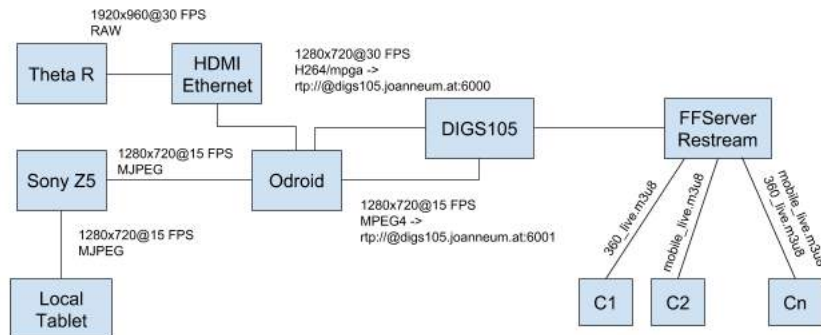


Abbildung 4.9: In dieser Abbildung wurden einerseits die Komponenten ausgewählt, die das Videomaterial aufnehmen, andererseits auch das Übertragungsprotokoll, sowie die einzelne Verteilung, wie der Livestream schlussendlich an die einzelnen Nutzer verteilt wird

dem HDMI Extender und dieser per Ethernet weiter zum ODROID. Der ODROID komprimiert das Video auf 1280x720 herunter und sendet dieses nun auf einen internen Joanneum Research Server weiter. Auf diesem Server wird nun mittels ffserver der Videostream für alle verbundenen Geräte bereitgestellt.

### Sony Z5 Videostream

Das Sony Z5 stellt einerseits einen direkten lokalen Stream für den Tabletcomputer bereit.

Um den Stream für alle anderen Geräte bereitstellen zu können, wird das Video mit einer Auflösung von 1280x720 und 15 FPS weiter an den ODROID gesendet. Dieser leitet nun den Videostream wieder auf den internen Server weiter und macht diesen nun mittels ffserver für alle verfügbar.

Der von uns verwendete ffserver stammt aus dem FFmpeg Projekt, welches die Konvertierung, Aufnahme, Streaming von verschiedenen Formaten ermöglicht.

## 4 Methodik

Durch diese von uns gewählte Variante, kann jedes Gerät, welches mit dem Internet verbunden ist, den Livestream empfangen, ob Mitarbeiter vor Ort, als auch die Mitarbeiter in der Einsatzzentrale und da diese die selben Bilder wie der Mitarbeiter im Einsatz bekommt, dadurch bei Notfällen auch schnell handeln.

- **Stabilisierungshardware**

Das chinesische Technologieunternehmen DJI stellt eine SDK, als auch eine Android Beispielsapp zur Verfügung mit der eine Anbindung sehr gut durchführbar ist.

Dadurch hat man die Möglichkeit per App, die Osmo Mobile Stabilisierungshardware zu steuern.

- **Steuerungshardware**

Nun geht es an die Implementierung und Einrichtung des Webservers am Smartphone.

Bei der Wahl der Webserverimplementation entschieden wir uns für die „nanohttpd“ Library <sup>13</sup>.

Dafür mussten wir ein html File erstellen und dieses File musste in der Android Application referenziert werden, damit nanohttpd dieses File auch finden konnte.

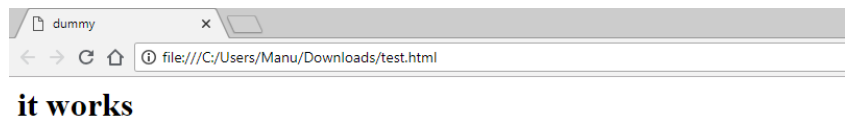


Abbildung 4.10: Testseite Webserver

Der nächste Schritt war das html File so zu verändern, damit unser Webinterface alle gewünschten Funktionen beinhaltet. Damit wir mit der Webseite auch interagieren konnten, musste noch ein JavaScript File angelegt werden, der diese Funktionen übernehmen soll.

---

<sup>13</sup><https://github.com/NanoHttpd/nanohttpd>

## 4 Methodik

Dafür muss eine Kommunikation zwischen der Webseite und dem Webserver auf dem Smartphone hergestellt werden und dies geschieht per HTTP GET Request.

Je nachdem welcher Interaktion vom User gewünscht wird, wird der jeweilige Request an den Webserver gesendet.

Dieser verarbeitet diesen Request, leitet die weiteren Schritte ein, die benötigt werden um den Request auszuführen und sendet einen Response an die Webseite zurück.

Dies kann zum Beispiel ein Bild, Positions- beziehungsweise Richtungsdaten oder auch der Link zum Livestream sein.

Die Webseite empfängt den Response, verarbeitet diesen und zeigt dem User das Ergebnis seiner Interaktion an.

In der Abbildung 4.11 sieht man nun die fertige Webseite, mit der die User nun interagieren kann.

In der Abbildung 4.12 sieht man die Webseite, mit der man die aktuelle Position auf der Karte, sowie den Neigungswinkel bildlich, als auch als Text sieht.

Die Seite ermöglicht einerseits das auslösen eines Fotos, als auch das Empfangen und ansehen des letzten aufgenommen Fotos

Gleichzeitig kann auch der Livestream gestartet, gestoppt, sowie auch angesehen werden.

Zusätzlich kann von der Webseite aus, auch die Stabilisierungshardware DJI in die verschiedenen Richtungen bewegt werden.

In der Abbildung 4.12 wird die Map angezeigt, dabei haben wir uns für Google Maps entschieden. Empfängt das Handy GPS Daten, wird in der Map die genaue Position als Punkt angezeigt.

Rechts sieht man den Neigungswinkel bildlich dargestellt, welche sich bei Bewegung des Smartphones in die jeweilige Richtung bewegt.

In der Abbildung 4.13 wird links das Bild angezeigt, welches entwe-

## 4 Methodik

der gerade aufgenommen oder als letztes Bild zurückgegeben wurde. Bei jedem aufgenommenen Foto wird zusätzlich eine Datei angelegt, die weitere Informationen des aufgenommenen Bildes beinhaltet. Beispielsweise den Zeitpunkt, Position und Richtungsdaten im Moment der Bildaufnahme.

Rechts wird der Livestream angezeigt, mit einem Klick auf den Link darüber, wird der Stream im Vollbildmodus angezeigt.

In der Mitte wird bei GPS Empfang verschiedene Daten wie Latitude, Longitude, Azimuth, sowie der Rollwert angezeigt.

In der Abbildung 4.14 gibt es dann noch Funktionen, damit der Nutzer gewisse Hardwarekomponenten über die Webseite steuern kann.

Einerseits gibt es die Möglichkeit ein Bild zu schießen, als auch das letzte aufgenommene Bild zu erhalten.

Weiteres kann der Livestream gestartet, als auch gestoppt werden.

Zusätzlich kann auch noch die Stabilisierungshardware DJI per Webseite gesteuert werden, um diese nach links, rechts, oben und runter zu steuern.

Neben der Möglichkeit der Interaktion über die Webseite, wollten wir auch noch weitere Möglichkeiten anbieten, falls ein System ausfallen sollte. Einerseits kann der HTTP GET Request direkt ohne der Webseite versendet werden um bestimmte Befehle auszuführen, außerdem wurde noch ein Google Cloud Messaging implementiert. Nachdem das Handy autorisiert wurde und man dafür einen Schlüssel generiert hat, kann man bestimmte Nachrichten an dieses Handy senden und dadurch bestimmte Befehle ausführen.

## 4 Methodik

Tabelle 4.3: Übersicht aller Kommandos

Name	Befehl	Beschreibung
Status des Gerätes abrufen	<code>http://digs105.joanneum.at:8080/pmt-server/status?deviceName=&lt;deviceName&gt;</code>	In der Abbildung 4.15 wird einerseits die IP Adresse des Servers angezeigt, als auch der Batterielevel, der freie Speicherplatz am Smartphone. Zusätzlich wird noch angezeigt, ob der Server momentan läuft oder ob am Smartphone gerade der Live-stream aktiv ist.
Bild auslösen und GeoJson File zurücksenden	<code>http://digs105.joanneum.at:8080/pmt-server/capture?deviceName=&lt;deviceName&gt;</code>	Falls dieser Befehl erfolgreich ist, wird ein Link zum Bild, sowie der Geoinformation am Server retourniert.
Permanentes Hochladen der GeoJson Informationen starten	<code>http://digs105.joanneum.at:8080/pmt-server/live_transmit_geojson?deviceName=&lt;deviceName&gt;</code>	Falls dieser Befehl erfolgreich ist, wird ein Link zum File retourniert.
Permanentes Hochladen der GeoJson Informationen stoppen	<code>http://digs105.joanneum.at:8080/pmt-server/stop_transmit_geojson?deviceName=&lt;deviceName&gt;</code>	Das permanente Hochladen wird gestoppt.
Videostreaming am Handy aktivieren	<code>http://digs105.joanneum.at:8080/pmt-server/start_streaming?deviceName=&lt;deviceName&gt;</code>	Hierbei wird der Videolivestream gestartet und die URL davon retourniert.
Videostreaming am Handy deaktivieren	<code>http://digs105.joanneum.at:8080/pmt-server/stop_streaming?deviceName=&lt;deviceName&gt;</code>	Mit diesem Befehl wird der Videolive-stream gestoppt.

### 4.1.3 Tests am Frequency

Das Frequency Festival <sup>14</sup> ist ein Musikfestival, welches jährlich seit 2001 stattfindet und 2009 von Salzburg nach St. Pölten übersiedelt ist. Im Jahr 2017 kamen 140.000 Menschen, daher war dies auch eine perfekte Möglichkeit unser Mastsystem zu testen.

Untergebracht waren wir in diesen Containerbüros (Abbildung 4.16 und Abbildung 4.17), in dem mehrere Monitore mit Livestreams der Veranstaltungsbereiche angezeigt wurden, außerdem waren genügend Steckdosen und Platz vorhanden um unser Equipment aufbauen zu können.

Damit der Masten auch gut und vor allem sicher auf dem Quad montiert werden kann, wurde uns von der Securityfirma ein Quad zur Verfügung gestellt, bei der eine zusätzliche Vorrichtung installiert wurde, in dem wir den Masten problemlos befestigen konnten (Abbildung 4.18).

Damit auch die benötigte Hardware kompakt und übersichtlich mitgenommen werden kann und während der Fahrt mit dem Quad auch nicht auseinander fällt, beziehungsweise die einzelnen Hardwarkomponenten sich im kompletten Koffer verteilen, wurden die einzelnen Teile mit einem Klettband zum Aufkleben befestigt.

Dadurch ist die Hardware im Koffer gut befestigt, die Komponenten können aber auch wieder einfach entfernt werden, beziehungsweise auch wieder sehr einfach befestigt werden, wie in der Abbildung 4.19 ersichtlich.

Im normalen, fahrenden Betrieb wird der Masten auf ca. ein bis zwei Metern ausgezogen, da bei größeren Höhen der Masten zu sehr wackeln würde.

Im stehenden Betrieb, kann der Masten aber bis zu sieben Metern ausgezogen werden und man bekommt dadurch einen sehr guten Überblick vom gesamten Bereich, wie man in der Abbildung 4.20 erkennen kann.

---

<sup>14</sup><https://www.frequency.at>

## 4 Methodik

Wie hier in der Abbildung 4.21 zu sehen, ist das Mastsystem auch für den Einsatz sehr gut geeignet.

Die App (Abbildung 5.14) an sich ist sehr einfach aufgebaut und nach dem Starten der App, können alle benötigten Funktionalitäten über den Webserver, beziehungsweise über das REST Service oder dem Google Cloud Messaging getätigt werden. Denn da das Smartphone am Masten montiert ist und dieses nicht ohne Grund entfernt werden soll, sollten die einzelnen Funktionalitäten, wie den Livestream zu starten/beenden oder Fotos aufnehmen nur bei möglichen Problemen mit dem Netzwerk verwendet werden.

Beim Start der App wird der WebServer aufgebaut und die IP Adresse mit dem Port angezeigt, auf der der WebServer zu finden ist.

Zusätzlich kann noch der Livestream gestartet, sowie gestoppt werden und Fotos aufgenommen werden.

Am Ende des Einsatzes kann mit dem Kill App Button die App und alle Hintergrundservices beendet werden.

### 4.1.4 Evaluierung der Tests

Bei den Tests des ersten Prototypen am Frequency in St. Pölten, konnten wir erstmals das System in einem realen Umfeld testen, um wirklich einen richtigen Eindruck zu bekommen, welche Aufgabenfelder das System lösen soll, damit es für die Sicherheitskräfte die bestmögliche Unterstützung bieten kann.

Außerdem konnten einige Punkte herausgefunden werden, die durch die Tests auf einer echten Großveranstaltung zum Vorschein gebracht worden sind.

Mobiles Internet

## 4 Methodik

Wir starteten den Aufbau und Initialisierung der Hard- und Softwarekomponenten am Tag o des Festivals.

An diesem Tag war die Besucheranzahl noch um einiges geringer, als an den weiteren Tagen. Wir testeten an diesem Tag o auch die Livestreamübertragung und die Ergebnisse waren dabei gut.

Durch die höhere Anzahl an Menschen, die das Festival in den weiteren Tagen besuchten, wurde leider auch die Bildübertragungsrate des Livestreames geringer, was ein Problem darstellt, da das Video nicht mehr in der Qualität übertragen werden konnte, wie gewünscht.

Es werden aufgrund der großen Anzahl der Besucher, externe Mobilfunkmasten der einzelnen Hersteller aufgestellt, jedoch sind diese bei der großen Auslastung nicht dafür geeignet, große Bilddaten zu versenden.

Wir fanden außerdem heraus, dass die Auslastung in der Früh um einiges geringer ist, daher konnten wir in der Früh auch bessere Ergebnisse erzielen, wie am restlichen Tag.

Neben dem LTE Internet von A1, testeten wir unser System auch mit T-Mobile LTE Internet, was aber zu keinem großen Unterschied geführt hat.

### ODROID

Damit es auch möglich ist die Livestream Daten der Smartphonekamera, als auch der der 360 Grad Kamera gleichzeitig auf den Server zu streamen, sollten diese Daten vorher vom ODROID konvertiert und daraufhin weitergesendet werden.

Neben der schon oben beschriebenen Problematik mit der zu geringen Bandbreite, war auch der ODROID mit der Konvertierung dieser zwei Livestreams komplett ausgelastet, dies führte dazu, dass sowohl der Stream der Handykamera, als auch der der 360 Grad Kamera nicht wie gewünscht verwendbar waren.

### 360 Grad Video



## 4 Methodik

Wir testeten Aufnahmen der 360 Grad Video mit einer Virtual Reality Brille, da diese auch nur damit Sinn machen um einen kompletten Überblick der Situation zu bekommen.

Jedoch führte einerseits die schnelle Bewegung durch den Quad und andererseits die Bewegung des Bildes durch die Kopfbewegungen zu Übelkeit und Schwindelgefühl.

### Osmo DJI

Bei der Stabilisierungshardware Osmo DJI muss das Smartphone darin optimal austariert befestigt werden, damit die Leistung auch 100% ausgenutzt werden kann.

Sollte dies nicht der Fall sein, kann es schon bei leichten Unebenheiten auf der Straße zu derartigen Problemen führen, dass sich die Osmo DJI, 90 Grad nach unten fallen lässt und auf dieser Position bleibt.

Dies bedeutet einen zeitaufwändigen Aufwand vor jeder Fahrt, um das Smartphone perfekt in der Waage in die Stabilisierungshardware einbauen zu können.

Da die Sicherheitskräfte auf den Quad auch teilweise schneller und auch über unebene Wege, wie Schotterstraßen, Wiesen und Äcker fahren müssen, ist die Osmo DJI auch austariert nicht in der Lage, diese starken Unebenheiten auszugleichen und fällt wieder 90 Grad nach unten und nur ein Neustart des Gerätes setzt es wieder in die Waagrechte.

### Akku Sony Z5

Da unser Smartphone, das Sony Z5, im ersten Prototypen sehr viel Arbeit leisten muss, wie zum Beispiel GPS/IMU Daten regelmäßig aufzunehmen und auch weiterzusenden, einen Livestream beziehungsweise Fotos aufnehmen und diese weiterzusenden, als auch der Webserver der im Hintergrund läuft und Befehle annimmt, ist der Stromverbrauch sehr hoch.

Dies bedeutet gleichzeitig, dass der Akku maximal 60 bis 90 Minuten hält.

## 4.2 Prototyp 2

Nach der Evaluierung des ersten Prototypen wurde nun mit der Arbeit begonnen, die Probleme, die bei den Tests auftraten, in einem zweiten Prototypen zu lösen.

### 4.2.1 Hardwarerecherche

#### Osmo Zenmuse X5R

Da es bei der Osmo DJI bei schlechten Straßenverhältnissen zu Problemen kam, wurde eine neue Variante einer Stabilisierungshardware versucht, die Osmo Zenmuse X5R (Abbildung 4.24).

Diese hat den Vorteil, dass die Kamera direkt verbaut ist, daher fällt auch das Austarieren davor weg und man kann sich sicher sein, dass die Kamera perfekt darauf abgestimmt ist, im Gegensatz wenn ein Smartphone darauf montiert wird.

#### Emlid Reach RTK

Da wir gerne einen Vergleich zwischen der GPS Positionierung des Smartphones und einer RTK Positionierung haben wollen, entschieden wir uns für diese Tests auf das Emlid Reach RTK.

Es ist im Gegensatz zu professionellen Real Time Kinematic Systemen günstiger und besitzt trotzdem eine Genauigkeit von wenigen Zentimetern (Abbildung 4.25).

---

<sup>15</sup><https://www.dji.com/zenmuse-x5r>

## 4.2.2 Softwareimplementierung

### Zugriff zu Zenmuse X5R

DJI bietet auf ihrer Homepage eine SDK <sup>17</sup> einerseits für iOS, aber auch für Android an. Mit einem Entwickleraccount, welchen man sehr einfach anlegen kann, gibt es die Möglichkeit die SDK herunterzuladen und mit der vorhandenen Dokumentation auf der DJI Seite diese auch problemlos in die App einzubinden.

Um eine Verbindung zur Zenmuse X5R herstellen zu können, muss vorher noch ein eindeutiger Schlüssel für die App auf der DJI Seite generiert werden, mit der man die App mit dem vorher generierten Schlüssel sich bei der Zenmuse registrieren kann.

Nun gibt die SDK uns eine Vielzahl an Methoden an um beispielsweise Fotos zu schießen, Videos aufzunehmen, die Ausrichtung zu ändern, den Status abzufragen und vieles mehr.

### Zugriff RTK

Um Zugriff zu den Daten von dem RTK System von Emlid zu erhalten, muss zuallererst das System mit Strom versorgt werden. Beim ersten Start beziehungsweise wenn das vordefinierte WLAN nicht gefunden wird, erstellt das Real Time Kinematic System einen Wi-Fi Hotspot.

Nun verbindet man sich auf einem Rechner/Handy/Tablet etc. mit diesem Hotspot. Geht man nun im Browser auf die Seite <http://192.168.42.1>, wird man zum ReachView Update weitergeleitet.

Hier, wie in der Abbildung 4.26, bekommt man nun die Möglichkeit, das RTK System mit dem gewünschten WLAN Netzwerk zu verbinden.

---

<sup>16</sup><https://docs.emlid.com/reach/>

<sup>17</sup><https://developer.dji.com/mobile-sdk/>

<sup>18</sup>[https://docs.emlid.com/reach/common/reachview/img/reachview/introduction/reach\\_view\\_updater\\_main.png](https://docs.emlid.com/reach/common/reachview/img/reachview/introduction/reach_view_updater_main.png)

Nach einem Neustart gelangt man nun auf die ReachView App Ansicht (Abbildung 4.27), bei der man nun alle Informationen geliefert bekommt, angefangen von Satellitenempfang, Positionsmodus, Position, etc.. Außerdem können hier auch alle weiteren Einstellungen vollzogen werden.

### 4.2.3 Tests

#### Mobiles Internet

Mit der zusätzlichen Einstellung die Qualität des Streames zu verändern und der optimierten Kodierung mit FFmpeg, konnte der Stream mit einer geringeren Bandbreite versendet werden und funktioniert somit auch bei starker Auslastung des Internets bei Großveranstaltungen.

#### Sony Z5 Akku

- Laufzeit Akku bei Leerlauf  
156 Minuten
- Durchgehendes Livestreaming + Capturen  
82 Minuten

#### DJI Akku

- Laufzeit Akku bei Arbeit
  - Durchgehende Videoaufnahme  
36 Minuten
  - Ständig in Bewegung mit Video + Fotoaufnahmen  
31 Minuten
- Laufzeit Akku bei Leerlauf  
43 Minuten

---

<sup>19</sup><https://docs.emlid.com/reach/common/reachview/img/reachview/introduction/reach.view.app.png>

- DJI Akkustand auslesen  
Mithilfe der SDK von DJI konnten verschiedene Statuswerte des Akkus, wie das Batteriewert, die aktuellen Volt/Ampere, als auch die Temperatur ausgelesen werden (Abbildung 4.28).

### 4.2.4 Evaluierung der Tests

#### Mobiles Internet

Aufgrund der Tatsache, dass mit FFmpeg die Bildübertragungsrate, sowie die Komprimierung optimiert wurde, sank auch die benötigte Bandbreite. Außerdem konnte die Qualität des Streames in der App geändert werden, sodass bei schlechter Netzabdeckung der Livestream, zwar mit etwas schlechterer Qualität, aber trotzdem problemlos funktioniert.

#### ODROID

Die Problematik, dass einerseits der Kamera Livestream, als auch der Stream der 360 Grad Kamera, gleichzeitig am ODROID komprimiert und weiter an den Server gesendet wird, könnte theoretisch mit dem Einbau zusätzlicher Hardware in den ODROID gelöst werden.

Da aber bei den Livetests, Schwindelgefühl und Übelkeit bei bewegenden 360 Grad Videos festgestellt wurde, fällt diese Problematik weg, da der 360 Grad Stream nur im Stand sinnvoll ist und der normale Stream nur im fahrenden Betrieb, fällt die Problematik mit dem gleichzeitigen Komprimieren von diesen zwei Videoquellen am ODROID weg.

#### Stabilisierungshardware

Da die neue Stabilisierungshardware, die Osmo Zenmuse X5R, die Kamera direkt verbaut hat, ist diese auch tarier und außerdem auch dafür ausgelegt auch bei größeren Unebenheiten problemlos funktionieren soll. Leider konnten bei den ersten Tests große Probleme mit dem Akku festgestellt

## 4 Methodik

werden.

Da die Fotos, sowie Videos, direkt auf dem Gerät komprimiert werden, ist dadurch der Akkuverbrauch sehr hoch.

### Akku Sony Z5

Da ein großer Teil der Arbeit, die das Handy beim ersten Prototyp noch verrichtet hat, nun auf zusätzliche Komponenten aufgeteilt wurde, hält nun der Akku lange genug und führt daher zu keinen Problemen mehr.

## 4 Methodik



Abbildung 4.11: In dieser Abbildung sieht man nun die fertige Webseite, mit der die jeweiligen User nun interagieren können

## 4 Methodik

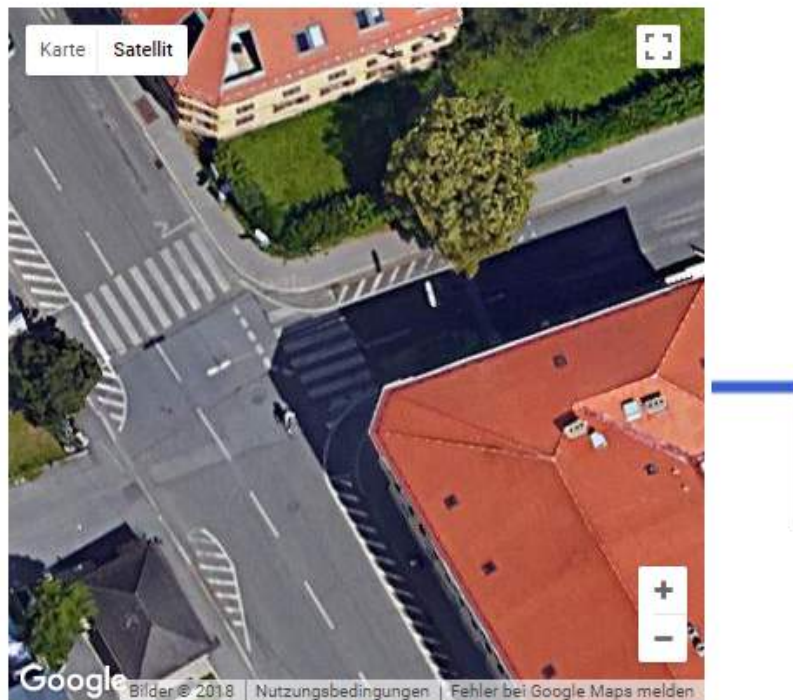


Abbildung 4.12: In der Abbildung wird unter anderem die Map angezeigt. Empfängt das Handy GPS Daten, wird in der Map die genaue Position als Punkt angezeigt. Rechts sieht man den Neigungswinkel bildlich dargestellt, welche sich bei Bewegung des Smartphones in die jeweilige Richtung bewegt.



## 4 Methodik



Abbildung 4.13: In der Abbildung wird links das Bild angezeigt, welches entweder gerade aufgenommen oder als letztes Bild zurückgegeben wurde. Bei jedem aufgenommen Foto wird zusätzlich eine Datei angelegt, die weitere Informationen des aufgenommenen Bildes beinhaltet. Beispielsweise den Zeitpunkt, Position und Richtungsdaten im Moment der Bildaufnahme. Rechts wird der Livestream angezeigt, mit einem Klick auf den Link darüber, wird der Stream im Vollbildmodus angezeigt. In der Mitte wird bei GPS Empfang verschiedene Daten wie Latitude, Longitude, Azimuth, sowie der Rollwert angezeigt.

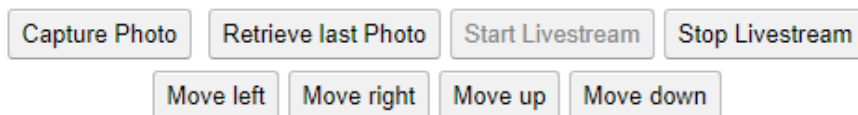


Abbildung 4.14: In der Abbildung gibt es dann noch Funktionen, damit der Nutzer gewisse Hardwarekomponenten über die Webseite steuern kann. Einerseits gibt es die Möglichkeit ein Bild zu schießen, als auch das letzte aufgenommene Bild zu erhalten. Weiteres kann der Livestream gestartet, als auch gestoppt werden. Zusätzlich kann auch noch die Stabilisierungshardware DJI per Webseite gesteuert werden, um diese nach links, rechts, oben und runter zu steuern.

## 4 Methodik

Server IP: 192.168.1.5, Battery level: 85%, Server status: running, Streaming status: not running, Free space: 486 MB

Abbildung 4.15: In der Abbildung wird einerseits die IP Adresse des Servers angezeigt, als auch der Batterielevel, sowie der freie Speicherplatz am Smartphone. Zusätzlich wird noch angezeigt, ob der Server momentan läuft oder ob am Smartphone gerade der Livestream aktiv ist.



Abbildung 4.16: Containerbüro am Frequency

## 4 Methodik



Abbildung 4.17: Containerbüro Bildschirme am Frequency

#### 4 Methodik



Abbildung 4.18: Damit der Masten auch gut und vor allem sicher auf dem Quad montiert werden kann, wurde uns von der Securityfirma ein Quad zur Verfügung gestellt, bei der eine zusätzliche Vorrichtung installiert wurde, in dem wir den Masten problemlos befestigen konnten.



#### 4 Methodik



Abbildung 4.19: Damit die benötigte Hardware auch kompakt und übersichtlich mitgenommen werden kann und während der Fahrt mit dem Quad auch nicht auseinander fällt beziehungsweise die einzelnen Hardwarkomponenten sich im kompletten Koffer verteilen, wurden die einzelnen Teile mit einem Klettband zum Aufkleben befestigt. Dadurch ist die Hardware im Koffer gut befestigt, die Komponenten können aber auch wieder einfach entfernt werden beziehungsweise auch wieder sehr einfach befestigt werden.

## 4 Methodik



Abbildung 4.20: Im stehenden Betrieb, kann der Masten aber bis zu sieben Metern ausgezogen werden und bekommt dadurch einen sehr guten Blick vom gesamten Bereich, wie man in der Abbildung erkennen kann.

#### 4 Methodik



Abbildung 4.21: Wie hier in der Abbildung zu sehen, ist das Mastsystem auch für den Einsatz sehr gut geeignet.

## 4 Methodik



Abbildung 4.22: Tablet App

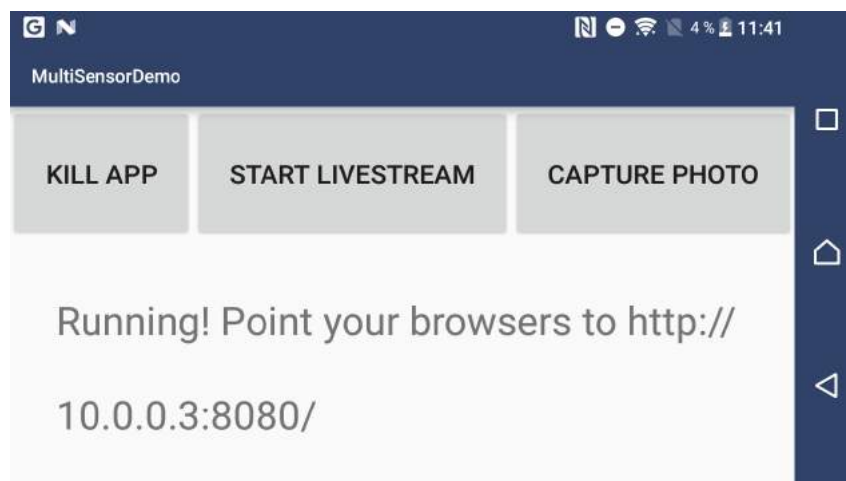


Abbildung 4.23: MutliSensor App



## 4 Methodik



Abbildung 4.24: Osmo Zenmuse X5R <sup>15</sup>



Abbildung 4.25: Emlid Reach RTK <sup>16</sup>

## 4 Methodik

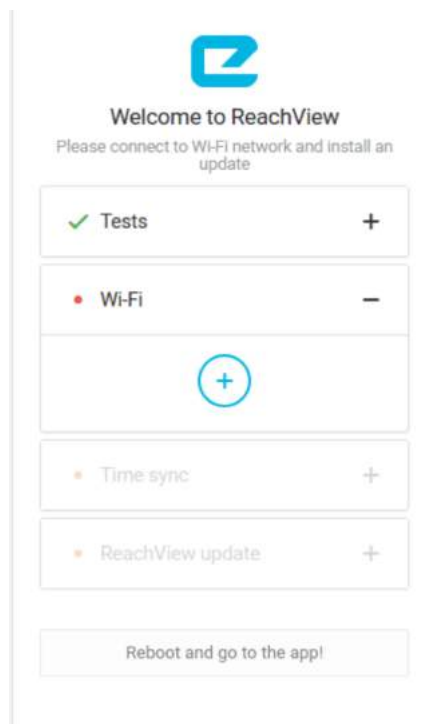


Abbildung 4.26: Nun bekommt man nun die Möglichkeit, das RTK System mit dem gewünschten WLAN Netzwerk zu verbinden.<sup>18</sup>

## 4 Methodik

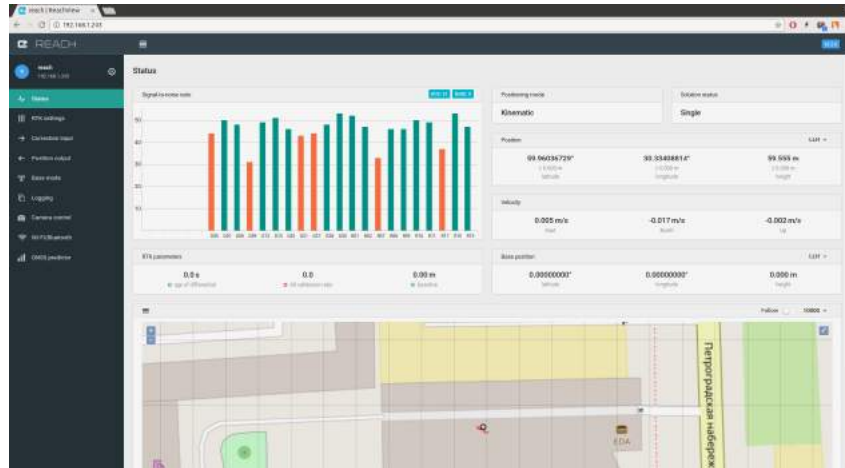


Abbildung 4.27: Nach einem Neustart gelangt man nun auf die ReachView App Ansicht, bei der man nun alle Informationen geliefert bekommt, angefangen von Satellitenempfang, Positionsmodus, Position, etc.. Außerdem können hier auch alle weiteren Einstellungen vollzogen werden. <sup>19</sup>

```
<message>Server IP: 192.168.8.104, Battery level: 56, Server status: running, Streaming status:
not running, Add orientation to bitmap: true, Free space: 3279.3828 MB, DJI
BatteryLevelPercent: 0 percent, DJI BatteryLevelPercent own: 87,39 percent, DJI Batterylevel:
1081, DJI Batterylevel Full: 1237, DJI CurrentVoltage: 11824 mV, DJI CurrentAmpere: -1579 mA,
DJI LifetimeRemaining: 100 Y, DJI Temperature: 34.700012 Grad Celsius</message>
```

Abbildung 4.28: Mithilfe der SDK von DJI konnten verschiedene Statuswerte des Akkus, wie das Batterielevel, die aktuellen Volt/ Ampere, als auch die Temperatur ausgelesen werden.

## 5 Ergebnis

Für die nachfolgenden Tests wurden verschiedene Android Smartphones, sowie ein professionelles GPS Gerät und ein RTK System verwendet.

Folgende Smartphones wurden verwendet:

- **Sony Z5** <sup>1</sup>

Das Sony Xperia Z5 ist ein High-End Smartphone und wurde von Sony entwickelt (Abbildung 5.1). Es wurde im September 2015 vorgestellt und kam im Oktober 2015 für den Einführungspreis von 699€ in den Verkauf.

Die neueste Android Version für das Sony Z5 ist Android 7.1 Nougat. Die satellitengestützte Ortsbestimmung werden durch die Ortungsmodul GPS, sowie GLONASS sichergestellt.

- **Samsung Galaxy S7** <sup>3</sup>

Das Samsung Galaxy S7 ist ebenso ein High-End Smartphone und wurde von Samsung entwickelt (Abbildung 5.2). Es wurde im Februar 2016 vorgestellt und kam im März 2016 für den Einführungspreis von 699€ in den Verkauf.

Die neueste Android Version für das Samsung Galaxy S7 ist Android 7.0 Nougat.

Die satellitengestützte Ortsbestimmung werden durch die Ortungsmodul

---

<sup>1</sup><http://www.aremobil.de/handys/5395-sony-xperia-z5/datenblatt>

<sup>2</sup><https://images-na.ssl-images-amazon.com/images/I/71Y-Lr2MZiL.SL1500-.jpg>

<sup>3</sup><http://www.samsung.com/at/smartphones/galaxy-s7/hardware/>

## 5 Ergebnis



Abbildung 5.1: Sony Z5 <sup>2</sup>

dule GPS, sowie GLONASS sichergestellt.



Abbildung 5.2: Galaxy S7 <sup>4</sup>

## 5 Ergebnis

- **Nexus 6** <sup>5</sup>

Das Nexus 6 ist ein High-End Smartphone und wurde von Google entwickelt (Abbildung 5.3). Es wurde im Oktober 2014 vorgestellt und kam im November 2014 für den Einführungspreis von 596€ in den Verkauf.

Die neueste Android Version für das Nexus 6 ist Android 7.1 Nougat. Die satellitengestützte Ortsbestimmung wird durch A-GPS sichergestellt.

- **Nexus 5** <sup>7</sup>

Das Nexus 5 ist ein High-End Smartphone und wurde von Google entwickelt (Abbildung 5.4). Es wurde im Oktober 2013 vorgestellt und kam im November 2013 für den Einführungspreis von 349€ in den Verkauf.

Die neueste Android Version für das Nexus 5 ist Android 6.0 Marshmallow.

Die satellitengestützte Ortsbestimmung werden durch die Ortungsmodule GPS, sowie GLONASS sichergestellt.

- **Moto G** <sup>9</sup>

Das Moto G (2. Generation) ist ein Mittelklasse Smartphone und wurde von Motorola entwickelt (Abbildung 5.5). Es kam im September 2014 für den Einführungspreis von 199€ in den Verkauf.

Die neueste Android Version für das Nexus 5 ist Android 6.0 Marshmallow.

Die satellitengestützte Ortsbestimmung wird durch das Ortungsmodule GPS sichergestellt.

---

<sup>4</sup>[https://images.samsung.com/is/image/samsung/global-mkt-galaxy-s7-overview-galaxy-s7-gallery\\_front\\_black](https://images.samsung.com/is/image/samsung/global-mkt-galaxy-s7-overview-galaxy-s7-gallery_front_black)

<sup>5</sup>[https://support.google.com/nexus/answer/6102470?hl=de&ref\\_topic=3415518](https://support.google.com/nexus/answer/6102470?hl=de&ref_topic=3415518)

<sup>6</sup><https://images-na.ssl-images-amazon.com/images/I/81nocFkBR%2BL..SL1500..jpg>

<sup>7</sup>[https://support.google.com/nexus/answer/6102470?hl=de&ref\\_topic=3415518](https://support.google.com/nexus/answer/6102470?hl=de&ref_topic=3415518)

<sup>8</sup><https://images-na.ssl-images-amazon.com/images/I/51HBaroPUoL.jpg>

<sup>9</sup><http://www.areamobile.de/handys/4315-motorola-moto-g-2-gen/datenblatt>

## 5 Ergebnis



Abbildung 5.3: Nexus 6 <sup>6</sup>

Als GPS Gerät wurde das Gerät von u-Blox verwendet.

Als RTK System fungiert das Emlid Reach RTK. Dieses System ist im Gegensatz zu professionellen Real Time Kinematic Systemen günstiger und besitzt trotzdem eine Genauigkeit von wenigen Zentimetern.

---

<sup>10</sup><https://www.specsraja.com/wp-content/uploads/2018/06/g2.jpg>

## 5 Ergebnis



Abbildung 5.4: Nexus 5 <sup>8</sup>



Abbildung 5.5: Moto G <sup>10</sup>



## 5.1 Vorbereitung

Es wurde eine Software geschrieben, die auf allen Android Geräten installiert wurde, welche nach einem bestimmten einstellbaren Zyklus, die Positionswerte, wie Latitude, Longitude, Höhe, sowie einem Zeitstempel ermittelt und diese werden dann mit dem Namen des Gerätes versehen (Abbildung 5.6).

Diese Informationen werden nun in eine csv (Comma-separated values) Datei geschrieben und am Handy gesichert (Abbildung 5.8).

Damit auch während der Durchführung der Tests die Werte kontrolliert werden können bzw. die Funktionalität der einzelnen Geräte überprüft werden können, wurde zusätzlich noch ein C# Programm erstellt, welches per TCP die Positionswerte und zusätzlich den Namen des sendenden Gerätes empfängt (Abbildung 5.7).

Dadurch kann versichert werden, dass auch alle Geräte funktionieren und die Positionswerte ermittelt werden können. Außerdem ist es mit dem C# Programm möglich, sämtliche empfangenen Daten aller Smartphones, ebenso als csv Datei, im selben Format wie auch die Datei direkt am Smartphone, abzuspeichern.

Dabei ist das Format so gewählt, dass zuerst der Name des Devices angezeigt wird, damit auch eindeutig zugeordnet werden kann, von welchem Gerät die GPS Daten kommen. Darauf folgen Latitude, Longitude, die Höhe, sowie ein Zeitstempel um die Zeit der GPS Aufnahme zu erhalten.

Der nächste Schritt war es sämtliche Vorbereitungen für das GPS Gerät zu vervollständigen.

Auf der Homepage von u-blox wird eine Software für Windows, aber auch für Android bzw. iOS angeboten.

Da die Smartphones schon für die GPS Tests benötigt wurden, entschieden wir uns für die Version für Windows, um dieses Programm am Windows

## 5 Ergebnis



Abbildung 5.6: Android App für GPS Daten

Tablet installieren zu können.<sup>11</sup>

Nachdem die Software installiert wurde, musste nur noch das Gerät per USB verbunden werden und wenn ein GPS Signal vorhanden ist, beginnt das Gerät die GPS Daten an das Programm zu senden.

Nun musste noch das RTK System von Emlid für die Tests vorbereitet

<sup>11</sup><https://www.u-blox.com/de/product/u-center-windows>

<sup>12</sup><https://docs.zubax.com/zubax-gnss/tutorials/u-center.png>

## 5 Ergebnis

Name	Latitude	Longitude	Altitude	Timestamp
Nexus 6	47.06461957	15.45244346	428.0	1525770305
Nexus 5	47.0646195	15.45244346	0.0	1525770357
Moto G	47.0641123	15.4524137	0.0	1525770370
Galaxy S7	47.06476247302	15.45253270702	421.540357605	1525770371
Sony Z5	47.06457906	15.45255801	417.0	1525770388
Nexus 5	47.06460341	15.45237148	452.0	1525770387
Nexus 6	47.06488708	15.45245441	427.0	1525770389
Galaxy S7	47.06476250547	15.45253210347	419.6280273516	1525770393
Moto G	47.06433021	15.4525413	0.0	1525770400
Galaxy S7	47.06476250547	15.45253210347	419.5838030541	1525770401
Nexus 5	47.06460337	15.45253143	470.0	1525770417
Sony Z5	47.0646035	15.45236226	417.0	1525770418
Galaxy S7	47.06476250547	15.45253210347	419.315109682	1525770423
Nexus 6	47.06488708	15.45245439	427.0	1525770425
Moto G	47.06547238	15.452388	494.0	1525770430
Galaxy S7	47.06476250547	15.45253210347	419.2553757135	1525770431
Nexus 5	47.06570816	15.45216074	476.0	1525770447
Sony Z5	47.06430441	15.45239344	416.0	1525770448
Galaxy S7	47.06476250547	15.45253210347	419.1411797874	1525770453
Moto G	47.06522365	15.45216069	473.0	1525770466
Nexus 6	47.06488708	15.45245475	427.0	1525770462
Galaxy S7	47.06476250547	15.45253210347	419.0944673607	1525770461
Nexus 5	47.06522365	15.45224189	465.0	1525770477
Sony Z5	47.0649008	15.45240762	417.0	1525770478
Galaxy S7	47.06476250547	15.45253210347	419.0084561027	1525770483

Abbildung 5.7: C# TCP Server für Positionsdatenempfang

```

Sony Z5;47.06461962;15.4519721;414.0;1525781351
Sony Z5;47.06460375;15.45196165;418.0;1525781381
Sony Z5;47.06462381;15.45198943;416.0;1525781411
Sony Z5;47.06462262;15.45198906;416.0;1525781441
Sony Z5;47.06462213;15.45198818;416.0;1525781472
Sony Z5;47.06462327;15.45198857;416.0;1525781502
Sony Z5;47.06465047;15.45198816;414.0;1525781532
Sony Z5;47.06463712;15.45197374;412.0;1525781562
Sony Z5;47.06463999;15.45197427;412.0;1525781592
Sony Z5;47.064641;15.45197505;411.0;1525781622
Sony Z5;47.06464275;15.45197609;411.0;1525781652
Sony Z5;47.06464362;15.45197625;411.0;1525781682
Sony Z5;47.06464377;15.45197627;411.0;1525781712

```

Abbildung 5.8: CSV Format am Handy

werden.

Um Zugriff zu den Daten von dem RTK System von Emlid zu erhalten, muss zuallererst das System mit Strom versorgt werden. Beim ersten Start bzw. wenn das vordefinierte WLAN nicht gefunden wird, erstellt das Real

## 5 Ergebnis

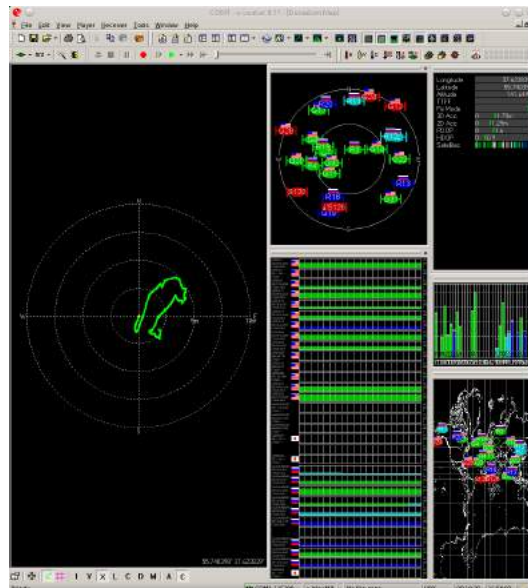


Abbildung 5.9: u-Center <sup>12</sup>

Time Kinematic System einen Wi-Fi Hotspot.

Nun verbindet man sich auf einem Rechner/Handy/Tablet etc. mit diesem Hotspot. Geht man nun im Browser auf die Seite <http://192.168.42.1>, wird man zum ReachView Update weitergeleitet. Hier bekommt man nun die Möglichkeit, das RTK System mit dem gewünschten WLAN Netzwerk zu verbinden.

Nach einem Neustart, wird man auf die Reach View App Seite weitergeleitet und kann dort nun alle weiteren Einstellungen vollziehen.

Damit auch der RTK Modus verwendet wird, muss dieser in den Einstellungen aktiviert werden. Diesen kann man aktivieren, indem man im Punkt RTK settings → Positioning mode auf kinematic stellt, wie hier in der Abbildung 5.10 ersichtlich.

Zusätzlich können noch weitere Einstellungen getätigt werden, wie zum

<sup>13</sup>[https://docs.emlid.com/reach/common/reachview/img/reachview/rtk\\_settings/rtk\\_set.png](https://docs.emlid.com/reach/common/reachview/img/reachview/rtk_settings/rtk_set.png)

## 5 Ergebnis

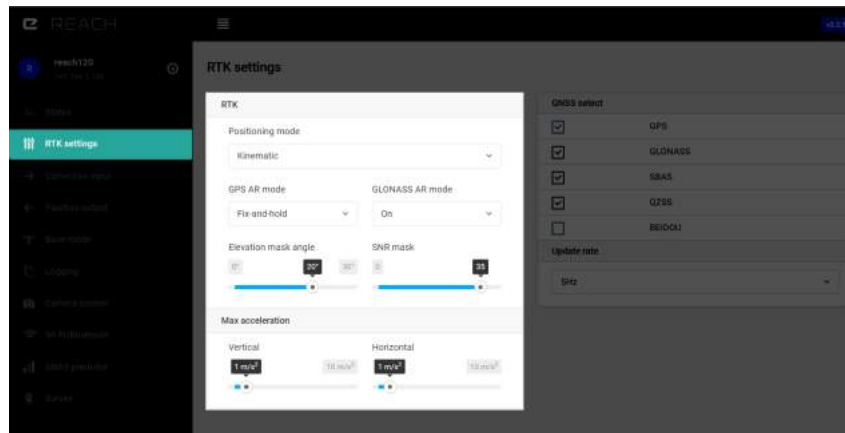


Abbildung 5.10: Damit auch der RTK Modus verwendet wird, muss dieser in den Einstellungen aktiviert werden. Diesen kann man aktivieren, indem man im Punkt RTK settings → Positioning mode auf kinematic stellt, wie hier in der Abbildung ersichtlich. <sup>13</sup>

Beispiel die Aktivierung/Deaktivierung von einzelnen GNSS Modulen (Globales Navigationssatellitensystem) (Abbildung 5.11).

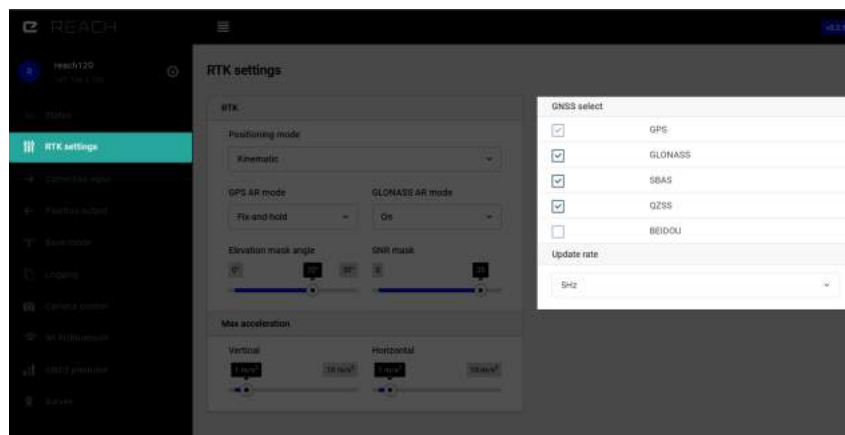


Abbildung 5.11: Zusätzlich können noch weitere Einstellungen getätigt werden, wie zum Beispiel die Aktivierung/Deaktivierung von einzelnen GNSS Modulen (Globales Navigationssatellitensystem). <sup>14</sup>

Damit das Reach Modul auch die Korrekturdaten von der Basistation be-

<sup>14</sup>[https://docs.emlid.com/reach/common/reachview/img/reachview/rtk\\_settings/gnss\\_sel.png](https://docs.emlid.com/reach/common/reachview/img/reachview/rtk_settings/gnss_sel.png)

## 5 Ergebnis

kommt, muss die IP Adresse, sowie der Port von der Basistation im Punkt Correction Input → TCP eingestellt werden.

Außerdem muss die Einstellung Client ausgewählt sein (Abbildung 5.12).

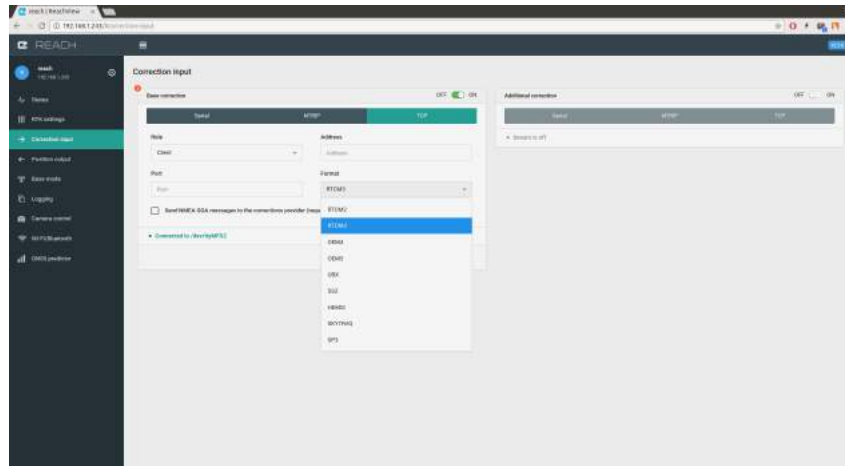


Abbildung 5.12: Damit das Reach Modul auch die Korrekturdaten von der Basistation bekommt, muss die IP Adresse, sowie der Port von der Basistation im Punkt Correction Input → TCP eingestellt werden. Außerdem muss die Einstellung Client ausgewählt sein. <sup>15</sup>

Nun muss bei der Basistation, beim Punkt Base Mode, der Korrekturausgang auf TCP gestellt sein und die als Rolle der Punkt Server ausgewählt werden (Abbildung 5.13).

Nun kann man im Punkt Status, die aktuelle Anzahl der empfangenen Satelliten, sowie deren Empfang sehen.

Außerdem werden die Punkte verschiedenfarbig in der Map eingezeichnet, je nachdem wie gut der Empfang ist (Abbildung 5.14).

Neben der verschiedenen Positionshardware wurde noch ein Windows Tablet, mit dem Namen Toughpad FZ-G1, verwendet und ein mobiles

<sup>15</sup>[https://docs.emlid.com/reach/common/reachview/img/reachview/correction\\_input/tcp.png](https://docs.emlid.com/reach/common/reachview/img/reachview/correction_input/tcp.png)

<sup>16</sup>[https://docs.emlid.com/reach/common/reachview/img/reachview/base\\_mode/output.png](https://docs.emlid.com/reach/common/reachview/img/reachview/base_mode/output.png)

<sup>17</sup>[https://docs.emlid.com/reach/img/reach/quickstart/reach\\_view\\_status\\_menu.png](https://docs.emlid.com/reach/img/reach/quickstart/reach_view_status_menu.png)

## 5 Ergebnis

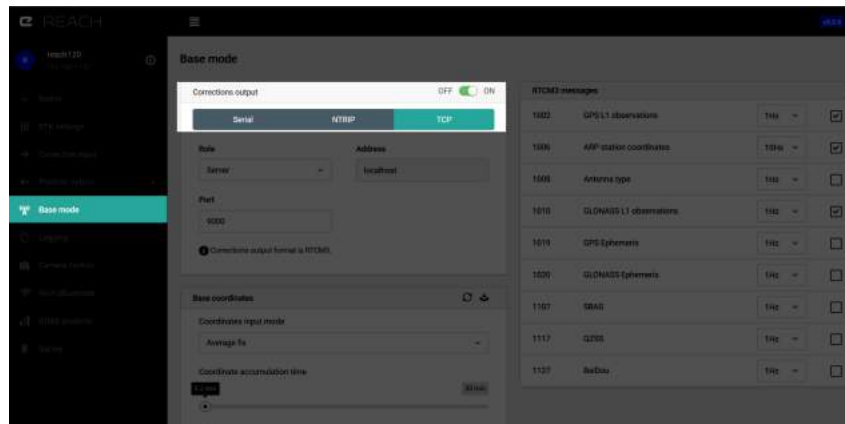


Abbildung 5.13: Nun muss bei der Basestation, beim Punkt Base Mode, der Korrekturausgang auf TCP gestellt sein und die als Rolle der Punkt Server ausgewählt werden.<sup>16</sup>

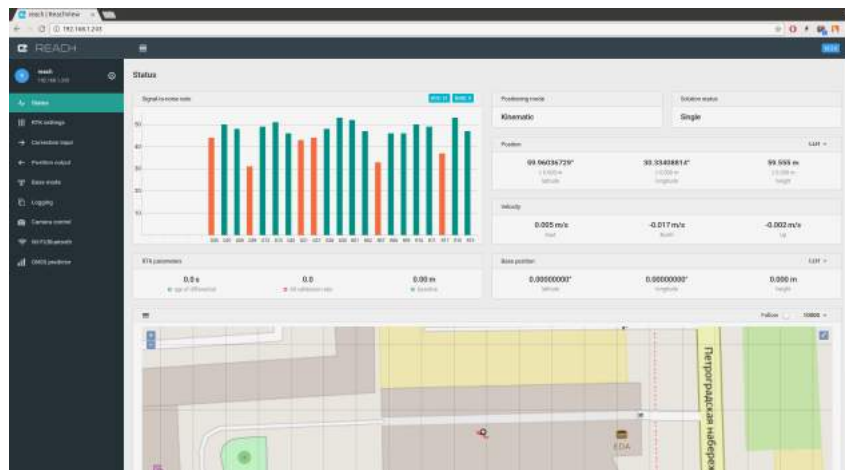


Abbildung 5.14: Nun kann im Punkt Status, die aktuelle Anzahl der empfangenen Satelliten, sowie deren Empfang sehen. Außerdem werden die Punkte verschiedenfarbig in der Map eingezeichnet, je nachdem wie gut der Empfang ist.<sup>17</sup>

Huawei Modem. Auf dem Windows Tablet wurde einerseits die Software des u-blox GPS Gerätes installiert um die Positionsdaten empfangen zu können.

## 5 Ergebnis

Das Toughpad FZ-G1 ist ein Tablet und sehr gut dafür geeignet, dieses im Freien einzusetzen, da es laut Herstellerangabe ein blendfreies Outdoor Display besitzt und zusätzlich noch einen Akku eingebaut hat, der bis zu 14 Stunden halten sollte <sup>18</sup>.

Außerdem wurde das C# Programm installiert um die Positionsdaten der verschiedenen Smartphones zu erhalten und zusätzlich wurde eine Verbindung zum RTK System aufgebaut, um die Funktionalitäten und Ergebnisse überprüfen zu können.

Damit auch die Hardware im selben Netzwerk ist und miteinander und auch mit dem Internet kommunizieren kann, wurde ein mobiles Modem der Firma Huawei verwendet.

Die Auswahl des Ortes für die Positionstests fiel auf das Gelände der Technischen Universität rund um die Steyrergasse, da dort vom Institut für Geodäsie ausgewählte Punkte genau vermessen wurden. Aus diesem Grund konnten genaue Referenzpunkte ermittelt werden, weshalb auch die Ergebnisse der Tests genauer ausfallen sollten.

Es wurde versucht verschiedene Punkte auszuwählen, einerseits Punkte die großteils frei von hohen Gebäuden sind, um auch einen besseren GPS Empfang zu erhalten und andererseits Punkte, die rundherum großteils von Gebäuden umgeben sind, um auch Referenzwerte für Einsätze in Großstädten zu erhalten.

In dieser Abbildung 5.15 wurden die verschiedenen Testpunkte eingezeichnet.

### 5.2 Durchführung der Tests

Die Durchführung der Tests wurde am 08. Mai 2018 in Graz durchgeführt.

---

<sup>18</sup><https://business.panasonic.de/computerloesungen/toughpad/fz-g1>



## 5 Ergebnis

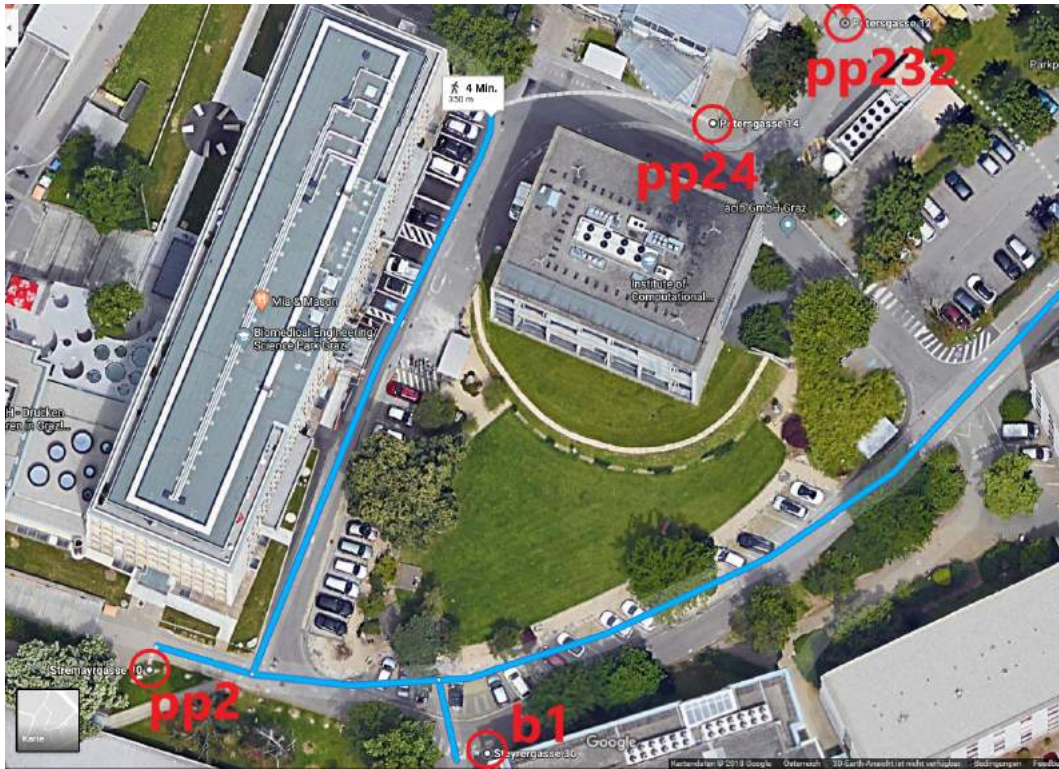


Abbildung 5.15: Es wurde versucht verschiedene Punkte auszuwählen, einerseits Punkte die großteils frei von hohen Gebäuden sind, um auch einen besseren GPS Empfang zu erhalten und andererseits Punkte, die rundherum großteils von Gebäuden umgeben sind, um auch Referenzwerte für Einsätze in Großstädten zu erhalten. In dieser Abbildung wurden die verschiedenen Testpunkte eingezeichnet.

- Referenzpunkt B1
- Referenzpunkt PP2
- Referenzpunkt PP24
- Referenzpunkt PP232

## 5 Ergebnis

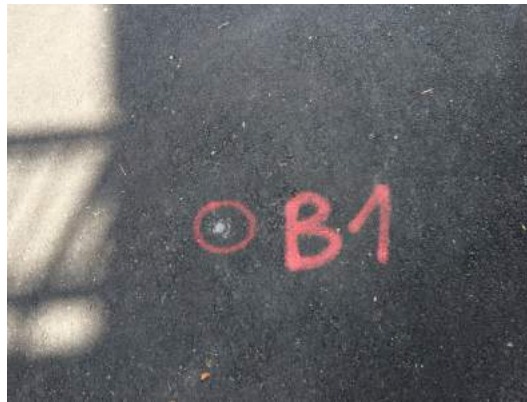


Abbildung 5.16: Referenzpunkt B1



Abbildung 5.17: Referenzpunkt B1 mit Hardware

### 5.3 Nachbereitung

Für die Nachbereitung der Daten wurde ein python Programm geschrieben, welche die aufgenommen Daten mittels der Matplotlib Library ploten soll, damit die Ergebnisse visuell gut sichtbar sind.

Da die gespeicherten GPS Daten der einzelnen Handys den selben Aufbau haben, konnten nun alle GPS Punkte aller teilgenommenen Smartphones für jeden Punkt in das Programm geladen werden und mit den Referenzpunkten der einzelnen Punkte verglichen werden.

## 5 Ergebnis



Abbildung 5.18: Referenzpunkt PP2



Abbildung 5.19: Referenzpunkt PP2 mit Hardware

Damit die GPS Daten des u-Blox Gerätes für mein python Programm verwendet werden können, wurde die Datei mit Hilfe der Webseite von GPS Visualizer <sup>19</sup> in das entsprechende Format konvertiert.

Die Daten des RTK Systems werden direkt am Gerät abgespeichert und können mit Hilfe der Weboberfläche, welche am RTK Gerät läuft, einfach heruntergeladen werden. Nachdem man die raw Daten des Rover und der Basis heruntergeladen hat, braucht man ein Programm, welches die GPS Nachbereitung erledigt um die Positionsdaten zu erhalten.

---

<sup>19</sup><http://www.gpsvisualizer.com/>

## 5 Ergebnis



Abbildung 5.20: Referenzpunkt PP24

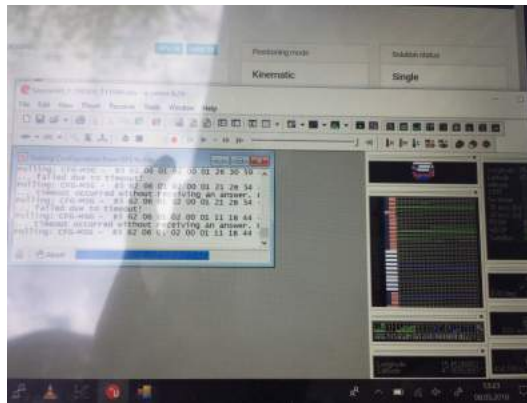


Abbildung 5.21: Professionelles GPS

Dafür wird das Programm RTKLIB<sup>20</sup> heruntergeladen, welches aus verschiedenen Einzelprogrammen besteht.

Online bietet Emlid nützliche Informationen bereit, wie die GPS Daten aufbereitet werden können.<sup>21</sup>

Zuerst wird das Programm RTKCONV benötigt, um aus dem UBX File des Rovers, unter anderem OBS und NAV Daten zu erhalten (Abbildung 5.29).

<sup>20</sup><https://files.emlid.com/RTKLIB/RTKLIB-Win-x64.zip>

<sup>21</sup><https://docs.emlid.com/reach/common/tutorials/gps-post-processing/>



## 5 Ergebnis



Abbildung 5.22: Referenzpunkt PP232



Abbildung 5.23: RTK Basistation

---

<sup>22</sup>[https://docs.emlid.com/reach/common/tutorials/img/reach/post-processing/rtkconv\\_format.png](https://docs.emlid.com/reach/common/tutorials/img/reach/post-processing/rtkconv_format.png)

## 5 Ergebnis



Abbildung 5.24: RTK Webseite

```
Samsung Galaxy S7;47.06468856179853;15.451996877647721;406.35765001620587;1525780398
Sony Z5;47.06465183;15.45204278;413.0;1525780399
Sony Z5;47.06465183;15.45204278;413.0;1525780399
MotoG;47.06472799;15.4520383;402.0;1525780403
MotoG;47.06472799;15.4520383;402.0;1525780403
Nexus 6;47.06467321;15.45202933;398.0;1525780404
Samsung Galaxy S7;47.06469003757825;15.451996859612878;406.3493345951135;1525780404
Nexus 6;47.06467321;15.45202933;398.0;1525780404
Sony Z5;47.06465176;15.45204245;413.0;1525780412
Sony Z5;47.06465176;15.45204245;413.0;1525780412
Nexus5;47.06467417;15.45206965;420.0;1525780421
Nexus5;47.06467417;15.45206965;420.0;1525780421
Samsung Galaxy S7;47.06469672724473;15.452000277802158;406.30363617904555;1525780428
```

Abbildung 5.25: GPS Beispelsdaten vom Referenzpunkt PP24

Bei den Optionen ist es noch wichtig, die aktuellste Version auszuwählen, bzw. alle benötigten Satellitensysteme zu aktivieren (Abbildung 5.30).

Das gleiche wird mit den Daten der Basisstation wiederholt und nach dem

<sup>23</sup>[https://docs.emlid.com/reach/common/tutorials/img/reach/post-processing/rtkconv\\_options.png](https://docs.emlid.com/reach/common/tutorials/img/reach/post-processing/rtkconv_options.png)

## 5 Ergebnis

```
#-----  
#Referenzpunkte  
  
#b1  
latFix = 47.0645559295478;  
lngFix = 15.45258851539731;  
heightFix = 359.5486;  
  
#pp2  
#latFix = 47.06464972890789;  
#lngFix = 15.452029800962412;  
#heightFix = 359.9385;  
  
#pp24  
#latFix = 47.06526616076544;  
#lngFix = 15.452963188331761;  
#heightFix = 360.2650;  
  
#pp232  
#latFix = 47.06538063441553;  
#lngFix = 15.453181301959098;  
#heightFix = 360.2410;  
  
#-----
```

Abbildung 5.26: Koordinaten der Referenzpunkte

Konvertieren erhält man nun die benötigten NAV, sowie OBS Daten.

Um nun die genaue Position aus Basis- und Roverstation zu erhalten, wird nun das Programm RTKPOST benötigt. Hier müssen nun die OBS Daten des Rovers und der Basisstation eingefügt werden, außerdem noch die NAV Daten des Rovers (Abbildung 5.31).

Bei den Optionen ist es wichtig als Positionsmodus Kinematic einzustellen um auch die genaue Position aus Basis- und Roverstation zu erhalten (Abbildung 5.32).

---

<sup>24</sup>[https://docs.emlid.com/reach/common/tutorials/img/reach/post-processing/rtkpost.adding\\_files.png](https://docs.emlid.com/reach/common/tutorials/img/reach/post-processing/rtkpost.adding_files.png)

## 5 Ergebnis

```
$GPGSV,7,1,25,02,41,073,34,06,15,037,19,12,49,085,31,14,25,283,*78
$GPGSV,7,2,25,24,18,158,30,25,78,339,24,29,56,229,30,31,33,309,21*75
$GPGSV,7,3,25,32,20,252,09,01,28,304,09,04,07,297,,09,02,344,*77
$GPGSV,7,4,25,26,21,199,33,27,,39,65,04,302,,66,08,356,*4A
$GPGSV,7,5,25,69,10,328,,73,53,124,28,74,67,333,15,75,14,316,*72
$GPGSV,7,6,25,80,01,133,,82,01,032,,83,49,040,35,84,66,190,20*7F
$GPGSV,7,7,25,85,13,207,21*47
$GPGSA,A,3,02,06,12,14,24,25,29,31,32,01,04,09,1.62,0.86,1.37*0B
$GPRMC,124454.11,A,4703.92673,N,01527.18864,E,0.168,0,080518,,A*6E
$GPVTG,0,T,,M,0.168,N,0.326,K,A*1B
$GPGGA,124455.13,4703.92677,N,01527.18864,E,1,13,0.86,361.8,M,42.4,M,,*50
$GPGLL,4703.92677,N,01527.18864,E,124455.13,A,*67
$GPGSV,7,1,25,02,41,073,29,06,15,037,19,12,49,085,32,14,25,283,*77
$GPGSV,7,2,25,24,18,158,31,25,78,339,31,29,56,229,31,31,33,309,20*70
$GPGSV,7,3,25,32,20,252,09,01,28,304,10,04,07,297,,09,02,344,*7F
$GPGSV,7,4,25,26,21,199,28,27,,38,65,04,302,,66,08,356,*41
$GPGSV,7,5,25,69,10,328,,73,53,124,28,74,67,333,15,75,14,316,*72
$GPGSV,7,6,25,80,01,133,,82,01,032,,83,49,040,37,84,66,190,22*7F
$GPGSV,7,7,25,85,13,207,21*47
$GPGSA,A,3,02,06,12,14,24,25,29,31,32,01,04,09,1.62,0.86,1.37*0B
$GPRMC,124455.13,A,4703.92677,N,01527.18864,E,0.074,0,080518,,A*65
$GPVTG,0,T,,M,0.074,N,0.144,K,A*11
$GPGGA,124456.05,4703.92682,N,01527.18867,E,1,12,0.86,361.8,M,42.4,M,,*5C
$GPGLL,4703.92682,N,01527.18867,E,124456.05,A,*6A
$GPGSV,7,1,25,02,41,073,29,06,15,037,20,12,49,085,21,14,25,283,*7F
$GPGSV,7,2,25,24,18,158,24,25,78,339,32,29,56,229,31,31,33,309,19*7D
$GPGSV,7,3,25,32,20,252,10,01,28,304,07,04,07,297,,09,02,344,*71
$GPGSV,7,4,25,26,21,199,26,27,,34,65,04,302,,66,08,356,*43
$GPGSV,7,5,25,69,10,328,,73,53,124,27,74,67,333,14,75,14,316,*7C
$GPGSV,7,6,25,80,01,133,,82,01,032,,83,49,040,37,84,66,190,20*7D
$GPGSV,7,7,25,85,13,207,21*47
$GPGSA,A,3,02,06,12,14,24,25,29,31,32,01,04,09,1.62,0.86,1.37*0B
$GPRMC,124456.05,A,4703.92682,N,01527.18867,E,0.191,0,080518,,A*62
$GPVTG,0,T,,M,0.191,N,0.371,K,A*1F
$GPGGA,124457.10,4703.92685,N,01527.18870,E,1,12,0.96,361.9,M,42.4,M,,*58
$GPGLL,4703.92685,N,01527.18870,E,124457.10,A,*6E
$GPGSV,7,1,25,02,41,073,29,06,15,037,20,12,49,085,24,14,25,283,*7A
$GPGSV,7,2,25,24,18,158,26,25,78,339,33,29,56,229,30,31,33,309,19*7F
$GPGSV,7,3,25,32,20,252,10,01,28,304,11,04,07,297,,09,02,344,*76
$GPGSV,7,4,25,26,21,199,24,27,,34,65,04,302,,66,08,356,*41
$GPGSV,7,5,25,69,10,328,,73,53,124,27,74,67,333,09,75,14,316,*70
$GPGSV,7,6,25,80,01,133,,82,01,032,,83,49,040,37,84,66,190,20*7D
$GPGSV,7,7,25,85,13,207,21*47
$GPGSA,A,3,02,06,12,14,24,25,29,31,32,01,04,09,1.77,0.96,1.49*07
$GPRMC,124457.10,A,4703.92685,N,01527.18870,E,0.194,0,080518,,A*63
$GPVTG,0,T,,M,0.194,N,0.377,K,A*1C
$GPGGA,124458.06,4703.92687,N,01527.18881,E,1,10,0.87,362.0,M,42.4,M,,*54
$GPGLL,4703.92687,N,01527.18881,E,124458.06,A,*6A
$GPGSV,7,1,25,02,41,073,35,06,15,037,19,12,49,085,36,14,25,283,*7E
$GPGSV,7,2,25,24,18,158,32,25,78,339,41,29,56,229,34,31,33,309,21*70
$GPGSV,7,3,25,32,20,252,10,01,28,304,09,04,07,297,,09,02,344,*7F
```

Abbildung 5.27: GPS U-Blox Beispieldaten

<sup>25</sup>[https://docs.emlid.com/reach/common/tutorials/img/reach/post-processing/rtpkpost.setting1\\_mode.png](https://docs.emlid.com/reach/common/tutorials/img/reach/post-processing/rtpkpost.setting1_mode.png)



## 5 Ergebnis



Abbildung 5.28: GPS Visualizer <sup>19</sup>

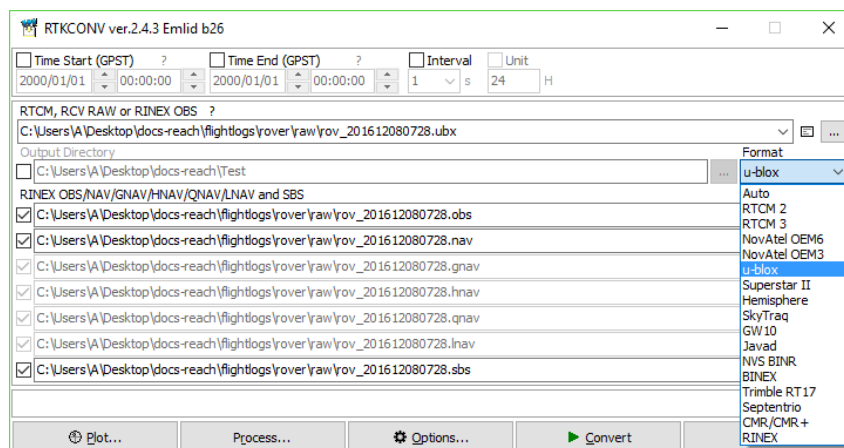


Abbildung 5.29: Zuerst wird das Programm RTKCONV benötigt, um aus dem UBX File des Rovers, unter anderem OBS und NAV Daten zu erhalten <sup>22</sup>

Nachdem alle benötigten Einstellungen vollzogen hat und die Konvertierung ausgeführt hat, bekommt man nun ein POS File.

Damit dieses File auch graphisch angezeigt werden kann, wird nun das Programm RTKPLOT benötigt.

Nachdem das File dort geöffnet wird, werden nun die aufgenommenen

## 5 Ergebnis

Options

RINEX Version: 3.03 (dropdown menu showing 2.10, 2.11, 2.12, 3.00, 3.01, 3.02, 3.03)

Station ID: 0000

☐ RINEX Name

RunBy/Obsv/A:

Comment:

Maker Name/#:

Rec #/Type/Vers:

Ant #/Type:

Approx Pos XYZ: ☐ 0.0000 0.0000 0.0000

Ant Delta H/E/N: 0.0000 0.0000 0.0000

☐ Scan Obs Types ☐ Half Cyc Corr ☐ Iono Corr ☐ Time Corr ☐ Leap Sec

Satellite Systems: ☒ GPS ☒ GLO ☒ GAL ☒ QZS ☒ SBS ☒ BDS ☒ IRN

Excluded Satellites:

Observation Types: ☒ C ☒ L ☒ D ☒ S

Frequencies: ☒ L1 ☒ L2 ☐ L5/3 ☐ L6 ☐ L7 ☐ L8 ☐ L9 Mask...

Option: Debug: OFF (dropdown menu) OK Cancel

Abbildung 5.30: Bei den Optionen ist es noch wichtig, die aktuellste Version auszuwählen, bzw. alle benötigten Satellitensysteme zu aktivieren.<sup>23</sup>

Punkte angezeigt, außerdem kann man diese Punkte nach fix, float oder single Punkten filtern (Abbildung 5.33).

<sup>26</sup><https://docs.emlid.com/reach/common/tutorials/img/reach/post-processing/rtkplot.pos.png>

## 5 Ergebnis

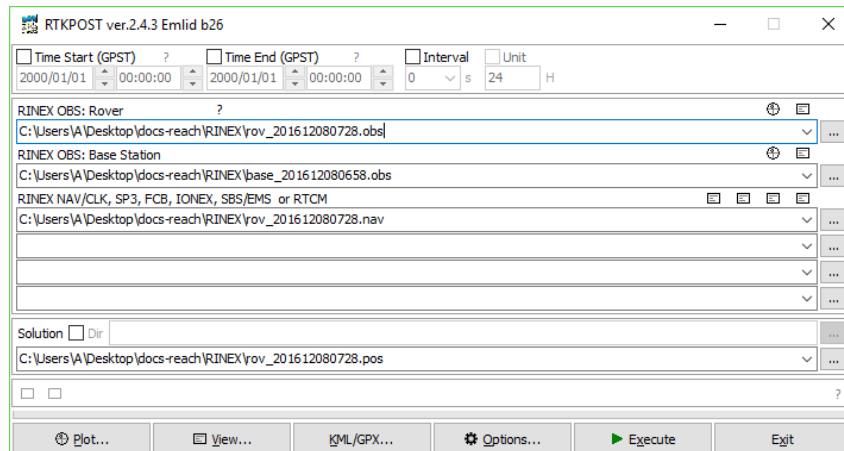


Abbildung 5.31: Um nun die genaue Position aus Basis- und Roverstation zu erhalten, wird nun das Programm RTKPOST benötigt. Hier müssen nun die OBS Daten des Rovers und der Basisstation eingefügt werden, außerdem noch die NAV Daten des Rovers. <sup>24</sup>

## 5.4 Testergebnisse

### 5.4.1 Positionsgenauigkeit Smartphone GPS Sensoren

Nun wurden die einzelnen Positionsdaten für die verschiedenen Referenzpunkte ausgewertet und analysiert.

- **Referenzpunkt B1**

Der erste Referenzpunkt B1 ist von Gebäuden umschlossen und daher liegt die Genauigkeit der Smartphones zwischen 5 und 34 Metern. Dabei erlangt das Sony Z5 mit einer Positionsgenauigkeit von ca. 5 Metern die besten und das Moto G die schlechtesten Werte von ca. 34 Metern (Abbildung 5.34).

- **Referenzpunkt PP2**

Der Referenzpunkt PP2 wird nicht von Gebäuden umschlossen und

## 5 Ergebnis

Tabelle 5.1: Übersicht der Positionsgenauigkeit in Metern von allen getesteten Smartphones vom Referenzpunkt B1. Damit die Durchschnittswerte aussagekräftiger sind, wurden die Positionswerte von den ersten Minuten, aufgrund der ungenauen Positionierung, sowie extreme Ausreißer nach oben, ignoriert.

Handy	Min	Platz	Max	Platz	Durchschnitt	Platz	Endresultat
Sony Z5	5.06	1.	16.34	1.	10.12	1.	<b>1.</b>
Galaxy S7	6.67	2.	19.12	2.	11.78	2.	<b>2.</b>
Nexus 6	10.44	3.	21.98	3.	12.39	3.	<b>3.</b>
Nexus 5	10.94	4.	27.80	4.	15.07	4.	<b>4.</b>
Moto G	17.45	5.	34.51	5.	19.95	5.	<b>5.</b>

Tabelle 5.2: Übersicht der Positionsgenauigkeit in Metern von allen getesteten Smartphones vom Referenzpunkt PP2. Damit die Durchschnittswerte aussagekräftiger sind, wurden die Positionswerte von den ersten Minuten, aufgrund der ungenauen Positionierung, sowie extreme Ausreißer nach oben, ignoriert.

Handy	Min	Platz	Max	Platz	Durchschnitt	Platz	Endresultat
Sony Z5	0.80	1.	11.10	4.	4.32	1.	<b>1.</b>
Galaxy S7	1.81	2.	11.69	5.	8.56	5.	<b>4.</b>
Nexus 6	3.00	4.	6.91	2.	4.75	3.	<b>3.</b>
Nexus 5	2.85	3.	6.01	1.	4.56	2.	<b>1.</b>
Moto G	5.14	5.	9.39	3.	8.09	4.	<b>4.</b>

## 5 Ergebnis

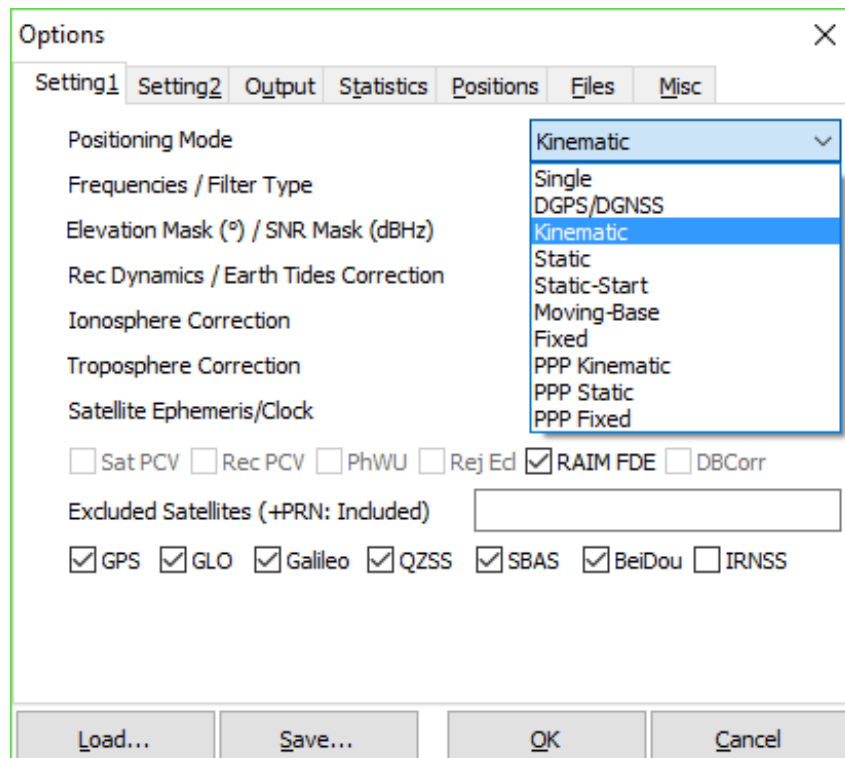


Abbildung 5.32: Bei den Optionen ist es wichtig als Positionsmodus Kinematic einzustellen um auch die genaue Position aus Basis- und Roverstation zu erhalten. <sup>25</sup>

daher steigt die Genauigkeit der Smartphones auch auf 0.8 bis 5 Metern. Dabei erlangt das Sony Z5 mit einer Positionsgenauigkeit von ca. 0.8 Metern die besten und das Moto G die schlechtesten Werte von ca. 5 Metern (Abbildung 5.35).

- **Referenzpunkt PP24**

Die Genauigkeit der Smartphones am Referenzpunkt PP24 liegt zwischen 0.7 und 7 Metern. Dabei erlangt das Samsung Galaxy S7 mit einer Positionsgenauigkeit von ca. 0.7 Metern die besten und das Moto G die schlechtesten Werte von ca. 7 Metern (Abbildung 5.36).

- **Referenzpunkt PP232**

## 5 Ergebnis

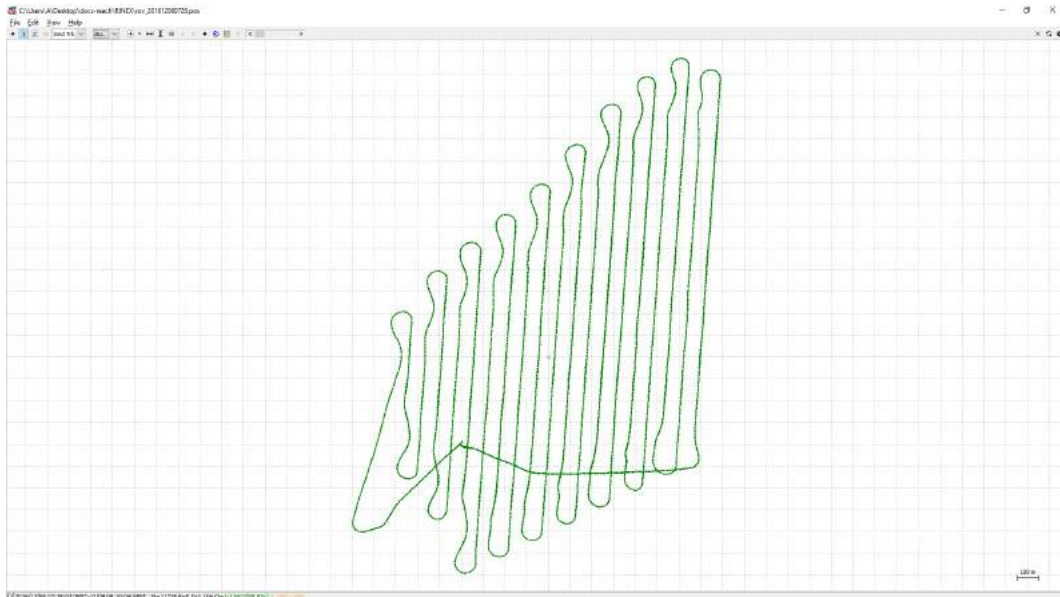


Abbildung 5.33: Nachdem das File dort geöffnet wird, werden nun die aufgenommenen Punkte angezeigt, außerdem kann man diese Punkte nach fix, float oder single Punkten filtern. <sup>26</sup>

Die Genauigkeit der Smartphones am Referenzpunkt PP232 liegt zwischen 1.3 und 5.3 Metern. Dabei erlangt das Sony Z5 mit einer Positionsgenauigkeit von ca. 1.3 Metern die besten und das Moto G die schlechtesten Werte von ca. 5.3 Metern (Abbildung 5.37).

### 5.4.2 Positionsgenauigkeit Professionellen GPS Sensor

- **Referenzpunkt B1**  
Die Positionsgenauigkeit des GPS Gerätes von u-Blox lag am Referenzpunkt B1 bei ca. 3 Metern (Abbildung 5.38).
- **Referenzpunkt PP2**  
Die Positionsgenauigkeit des GPS Gerätes von u-Blox lag am Referenzpunkt PP2 bei ca. 2 Metern (Abbildung 5.39).

## 5 Ergebnis

Tabelle 5.3: Übersicht der Positionsgenauigkeit in Metern von allen getesteten Smartphones vom Referenzpunkt PP24. Damit die Durchschnittswerte aussagekräftiger sind, wurden die Positionswerte von den ersten Minuten, aufgrund der ungenauen Positionierung, sowie extreme Ausreißer nach oben, ignoriert.

Handy	Min	Platz	Max	Platz	Durchschnitt	Platz	Endresultat
Sony Z5	1.74	2.	6.43	3.	3.04	1.	<b>1.</b>
Galaxy S7	0.75	1.	7.44	4.	3.16	2.	<b>2.</b>
Nexus 6	2.05	3.	4.78	1.	3.34	3.	<b>2.</b>
Nexus 5	5.39	4.	5.87	1.	5.56	4.	<b>4.</b>
Moto G	7.13	5.	8.52	3.	7.68	5.	<b>5.</b>

Tabelle 5.4: Übersicht der Positionsgenauigkeit in Metern von allen getesteten Smartphones vom Referenzpunkt PP232. Damit die Durchschnittswerte aussagekräftiger sind, wurden die Positionswerte von den ersten Minuten, aufgrund der ungenauen Positionierung, sowie extreme Ausreißer nach oben, ignoriert.

Handy	Min	Platz	Max	Platz	Durchschnitt	Platz	Endresultat
Sony Z5	1.31	1.	2.97	1.	2.17	1.	<b>1.</b>
Galaxy S7	1.36	2.	5.98	4.	3.91	2.	<b>2.</b>
Nexus 6	4.03	4.	5.94	3.	5.46	4.	<b>4.</b>
Nexus 5	3.87	3.	5.92	2.	4.12	3.	<b>2.</b>
Moto G	5.34	5.	6.12	5.	5.68	5.	<b>5.</b>

## 5 Ergebnis

- **Referenzpunkt PP24**

Die Positionsgenauigkeit des GPS Gerätes von u-Blox lag am Referenzpunkt PP24 bei ca. 1.7 Metern (Abbildung 5.40).

- **Referenzpunkt PP232**

Die Positionsgenauigkeit des GPS Gerätes von u-Blox lag am Referenzpunkt PP232 bei ca. 5.2 Metern (Abbildung 5.41).

### 5.4.3 Positionsgenauigkeit RTK System

Dabei wurden zwei verschiedene Varianten untersucht. Einerseits wurden die Positionsdaten nur aus der einzelnen RTK Roverstation heran genommen, andererseits die GPS Daten aus der Kombination zwischen der Basisstation und der Roverstation, bei der sich ein fixer Zustand der Position hergestellt hat.

- **Referenzpunkt B1 Kombination**

Die Positionsgenauigkeit des RTK Gerätes von Emlid lag am Referenzpunkt B1 bei weniger als einen Millimeter. (Abbildung 5.42).

- **Referenzpunkt B1 Roverstation**

Die Positionsgenauigkeit des RTK Gerätes von Emlid lag am Referenzpunkt B1 im besten Fall bei ca. 14 Zentimetern. (Abbildung 5.43).

- **Referenzpunkt PP2 Kombination**

Die Positionsgenauigkeit des RTK Gerätes von Emlid lag am Referenzpunkt PP2 bei ca. 1.8 Millimetern (Abbildung 5.44).

- **Referenzpunkt PP2 Roverstation**

Die Positionsgenauigkeit des RTK Gerätes von Emlid lag am Referenzpunkt PP2 im Besten Fall bei ca. 7 Zentimetern (Abbildung 5.45).



## 5 Ergebnis

- **Referenzpunkt PP24 Kombination**  
Die Positionsgenauigkeit des RTK Gerätes von Emlid lag am Referenzpunkt PP24 bei ca. 12 Zentimetern (Abbildung 5.46).
- **Referenzpunkt PP24 Roverstation**  
Die Positionsgenauigkeit des RTK Gerätes von Emlid lag am Referenzpunkt PP24 im Besten Fall bei ca. 13 Zentimetern (Abbildung 5.47).
- **Referenzpunkt PP232 Kombination**  
Die Positionsgenauigkeit des RTK Gerätes von Emlid lag am Referenzpunkt PP232 bei ca. 13 Millimetern (Abbildung 5.48).
- **Referenzpunkt PP232 Roverstation**  
Die Positionsgenauigkeit des RTK Gerätes von Emlid lag am Referenzpunkt PP232 im besten Fall bei ca. 6 Zentimetern (Abbildung 5.49).

### 5.4.4 Professionellen GPS Sensor vs. Smartphone GPS Sensoren vs. RTK System

Aufgrund der Tatsache, dass das RTK System mit einer Genauigkeit von wenigen Zentimetern bzw. Millimetern, ganz klar genauer ist als die restlichen GPS Systeme, wird nun das genaueste Handy, in diesem Fall das Sony Z5, mit dem GPS System von u-Blox, in den einzelnen Referenzpunkte verglichen.

- **Referenzpunkt B1**  
Am Referenzpunkt B1 sieht man, dass sich das u-Blox GPS System ca. 10 Minuten schneller eine gute Positionsgenauigkeit ermittelt, am Ende liegt die Genauigkeit jedoch sowohl beim Sony Z5, als auch beim u-Blox GPS System bei ca. 3-4 Metern (Abbildung 5.50).
- **Referenzpunkt PP2**  
Am Referenzpunkt PP2 liegt die Genauigkeit beim Sony Z5 bei ca.

## 5 Ergebnis

Tabelle 5.5: Übersicht der Positionsgenauigkeit in Metern von allen getesteten Geräten vom Referenzpunkt B1. Damit die Durchschnittswerte aussagekräftiger sind, wurden die Positionswerte von den ersten Minuten, aufgrund der ungenauen Positionierung, sowie extreme Ausreißer nach oben, ignoriert.

Gerät	Min	Platz	Max	Platz	Durchschnitt	Platz	Endresultat
Sony Z5	5.06	3.	16.34	3.	10.12	3.	3.
u-blox	1.52	2.	9.78	1.	4.71	2.	2.
RTK Rover	0.14	1.	13.17	2.	3.98	1.	1.

Tabelle 5.6: Übersicht der Positionsgenauigkeit in Metern von allen getesteten Geräten vom Referenzpunkt PP2. Damit die Durchschnittswerte aussagekräftiger sind, wurden die Positionswerte von den ersten Minuten, aufgrund der ungenauen Positionierung, sowie extreme Ausreißer nach oben, ignoriert.

Gerät	Min	Platz	Max	Platz	Durchschnitt	Platz	Endresultat
Sony Z5	0.80	2.	11.10	2.	4.32	2.	2.
u-blox	2.12	3.	13.18	3.	8.14	3.	3.
RTK Rover	0.07	1.	4.93	1.	2.92	1.	1.

0.8 Metern, beim u-Blox GPS System bei ca. 3 Metern (Abbildung 5.51).

- **Referenzpunkt PP24**

Am Referenzpunkt PP24 ist die Stabilität des GPS Empfanges beim Sony Z5 sehr gut. Die Positionsgenauigkeit beim Sony Z5 liegt bei ca. 3 Metern, beim u-Blox GPS System bei ca. 2 Metern (Abbildung 5.52).

- **Referenzpunkt PP232**

Am Referenzpunkt PP24 ist die Stabilität des GPS Empfanges beim Sony Z5 sehr gut. Die Positionsgenauigkeit beim Sony Z5 liegt bei ca. 2 Metern, beim u-Blox GPS System bei ca. 5 Metern (Abbildung 5.53).

## 5 Ergebnis

Tabelle 5.7: Übersicht der Positionsgenauigkeit in Metern von allen getesteten Geräten vom Referenzpunkt PP24. Damit die Durchschnittswerte aussagekräftiger sind, wurden die Positionswerte von den ersten Minuten, aufgrund der ungenauen Positionierung, sowie extreme Ausreißer nach oben, ignoriert.

Gerät	Min	Platz	Max	Platz	Durchschnitt	Platz	Endresultat
Sony Z5	1.74	3.	6.43	1.	3.04	1.	1.
u-blox	1.52	2.	9.78	2.	4.71	2.	2.
RTK Rover	0.13	1.	18.92	3.	6.11	3.	3.

Tabelle 5.8: Übersicht der Positionsgenauigkeit in Metern von allen getesteten Geräten vom Referenzpunkt PP232. Damit die Durchschnittswerte aussagekräftiger sind, wurden die Positionswerte von den ersten Minuten, aufgrund der ungenauen Positionierung, sowie extreme Ausreißer nach oben, ignoriert.

Gerät	Min	Platz	Max	Platz	Durchschnitt	Platz	Endresultat
Sony Z5	1.31	2.	2.97	1.	2.17	1.	1.
u-blox	5.44	3.	14.26	3.	11.10	3.	3.
RTK Rover	0.06	1.	9.67	2.	4.80	2.	2.

#### **5.4.5 Professionellen GPS Sensor vs. Smartphone Z5 Sensor vs. RTK Sensor Genauigkeitstest bei Bewegung**

Um auch zu testen, wie genau die Sensoren bei Bewegung sind, wurde eine Teststrecke mit den GPS Sensoren abgegangen um zu sehen, wie weit die Positionen der unterschiedlichen Sensoren auseinander liegen.

Dabei wurde festgestellt, wie in der Abbildung 5.54 ersichtlich, dass sich alle GPS Sensoren innerhalb von wenigen Metern zu der abgelaufenen Route befinden.

#### **5.4.6 Professionellen IMU Sensor vs. Smartphone IMU Sensoren vs. RTK System**

Bei den IMU Genauigkeitstests wurde ein professioneller IMU Sensor, nämlich der LPMS-B <sup>27</sup> von der Firma LP-Research verwendet. Nebenbei wurde noch das Sony Z5 und Samsung Galaxy S7 Smartphone benutzt und zusätzlich noch das RTK System von Emlid Reach.

Dabei wurden die Geräte mehrmals um eine bestimmte Gradanzahl im Kreis bewegt und dabei die yaw, pitch, sowie roll Werte entnommen. Diese Werte wurden mit den tatsächlichen Daten verglichen und aus allen Ergebnissen für jedes Gerät ein Durchschnittsgenauigkeitswert in Grad ermittelt.

Dabei erkennt man, dass der professionelle IMU Sensor LPMS-B die genauesten Werte liefert, dicht verfolgt von der IMU aus dem RTK System. Die zwei Smartphones hatten eine Genauigkeit von ca. einem Grad und lagen aus dem Grund hinter den beiden anderen getesteten IMU Sensoren.

---

<sup>27</sup><https://www.lp-research.com/lpms-b/>

## 5 Ergebnis

Tabelle 5.9: Übersicht IMU Genauigkeitstest von Professionellen IMU Sensor vs. Smartphone IMU Sensoren vs. RTK System. Dabei erkennt man, dass der professionelle IMU Sensor LPMS-B die genauesten Werte liefert, dicht verfolgt von der IMU aus dem RTK System. Die zwei Smartphones hatte eine Genauigkeit von ca. einem Grad und lagen aus dem Grund hinter den beiden anderen getesteten IMU Sensoren.

Gerät	Yaw	Platz	Pitch	Platz	Roll	Platz	Endresultat
Sony Z5	0.85	3	1.12	4.	1.23	3.	3.
Galaxy S7	0.93	4.	1.09	3	1.47	4.	4.
RTK Rover	0.27	2.	0.33	2	0.59	2.	2.
LPMS-B	0.09	1.	0.12	1.	0.28	1.	1.

### 5.5 Diskussion

Bei den Tests hat man gesehen, dass das RTK System, obwohl es eine low-cost Variante ist, mit einer Genauigkeit von wenigen Millimetern bzw. Zentimetern ganz klar die besten Werte aller getesteten GPS Sensoren erreichte.

Bei dem Vergleich zwischen dem professionellen GPS System und den Handy GPS Sensoren erkennt man, dass ältere Smartphones wie das Moto G nicht mit der Genauigkeit eines professionellen GPS Systemes mithalten können.

Wenn aber die neuen Smartphones, wie das Sony Z5, als auch das Samsung Galaxy S7 für die Tests hergenommen werden, sieht man, dass die GPS Sensoren dieser Smartphones gleich gute, teilweise sogar bessere Werte liefern, als das professionelle GPS System von u-blox.

Bei der Ergebnistabelle Tabelle 5.10 sieht man, dass sowohl das Sony Z5 und der RTK Rover ex aequo auf dem ersten Platz landeten, mit einer Durchschnittsplatzierung von 1.75. Den dritten Platz erreichte der u-blox Sensor, mit einer Durchschnittsplatzierung von 2.5.

Falls zentimetergenaue Positionsdaten benötigt werden, muss man auf ein

## 5 Ergebnis

Tabelle 5.10: Bei der Ergebnistabelle sieht man, dass sowohl das Sony Z5 und der RTK Rover ex aequo auf dem ersten Platz landeten, mit einer Durchschnittsplatzierung von 1.75. Den dritten Platz erreichte der u-blox Sensor, mit einer Durchschnittsplatzierung von 2.5.

Gerät	B1	PP2	PP24	PP232	Durchschnitt	Endresultat
Sony Z5	3.	2.	1.	1.	1.75	1.
u-blox	2.	3.	2.	3.	2.50	3.
RTK Rover	1.	1.	3.	2.	1.75	1.

RTK System setzen, welches immer genauer sein wird, als herkömmliche GPS Systeme, da die Fehlerquellen, beispielsweise das Wetter, das ein GPS System mitbringen, werden bei einem RTK System durch die Basisstation ausgeglichen.

Bei den IMU Tests Tabelle 5.9 erkennt man, dass der professionelle IMU Sensor LPMS-B der Genaueste unter den getesteten Geräten war. Dahinter folgt der RTK Rover, gefolgt von den zwei getesteten Smartphones, das Sony Z5 und das Samsung Galaxy S7 auf den Plätzen 3 und 4.

Durch die Tests der verschiedenen GPS- und IMU Sensoren, konnten wir wichtige Informationen über die Genauigkeit der einzelnen Geräte erhalten und somit die beste Variante für unser System finden. Aufgrund der Tatsache, dass unser System einerseits günstig und platzsparend sein sollte, entschieden wir uns für das Sony Z5.

Da dieses Smartphone von den getesteten Handys die höchste Genauigkeit bei den GPS Tests, als auch bei den IMU Tests lieferte und die Genauigkeit für unser Mastsystem absolut ausreichend ist.

Zusätzliche Sensoren, wie externe GPS bzw. IMU Geräte, würden einerseits die Komplexität des Systems stark erhöhen, andererseits auch den Preis.

## 5 Ergebnis

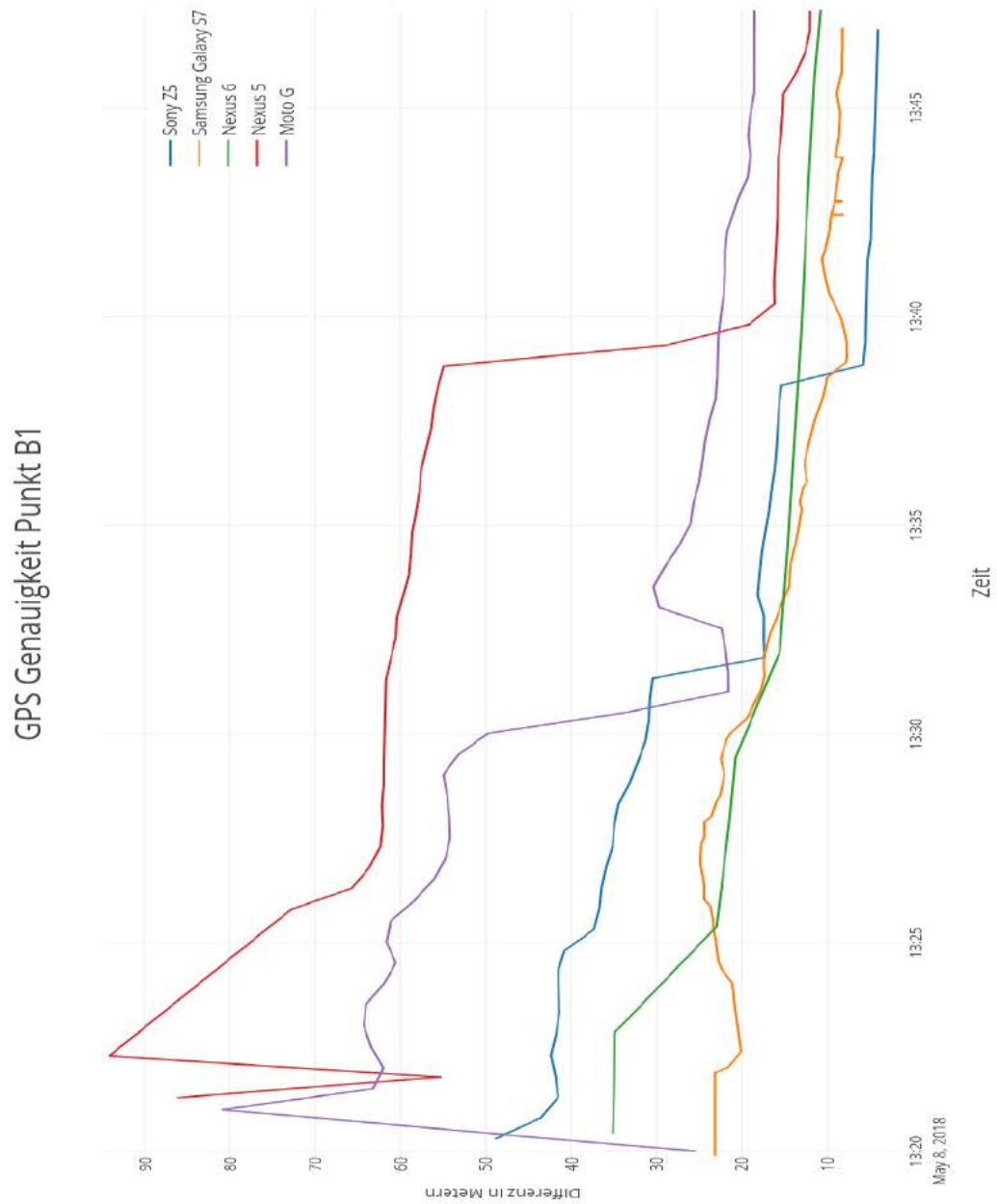


Abbildung 5.34: Der erste Referenzpunkt B1 ist von Gebäuden umschlossen und daher liegt die Genauigkeit der Smartphones zwischen 4 und 18 Metern. Dabei erlangt das Sony Z5 mit einer Positionsgenauigkeit von ca. 4 Metern die besten und das Moto G die schlechtesten Werte von ca. 18 Metern.

## 5 Ergebnis

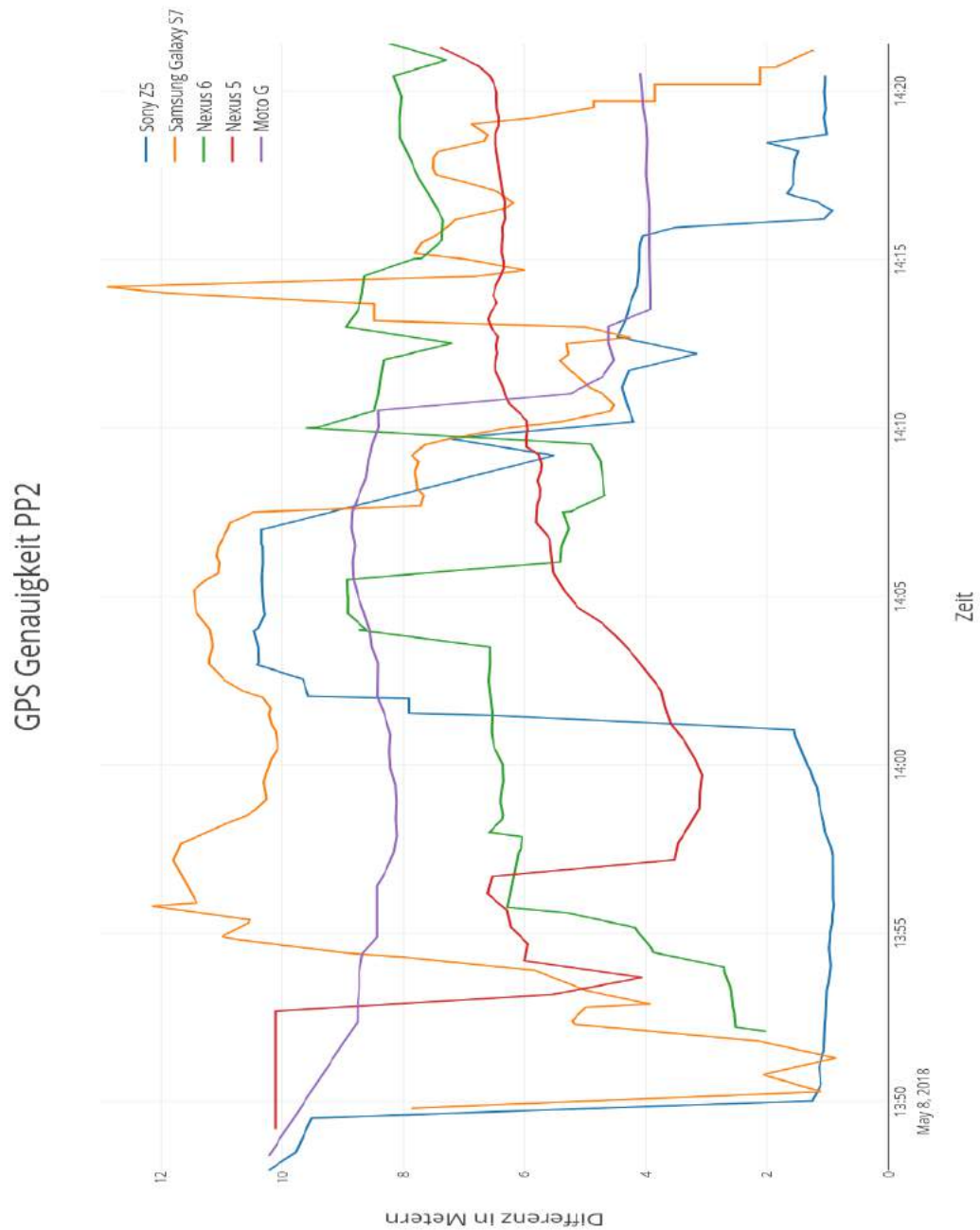


Abbildung 5.35: Der Referenzpunkt PP2 wird nicht von Gebäuden umschlossen und daher steigt die Genauigkeit der Smartphones auch auf 0.8 bis 4 Metern. Dabei erlangt das Sony Z5 mit einer Positionsgenauigkeit von ca. 0.8 Metern die besten und das Moto G die schlechtesten Werte von ca. 4 Metern.



## 5 Ergebnis

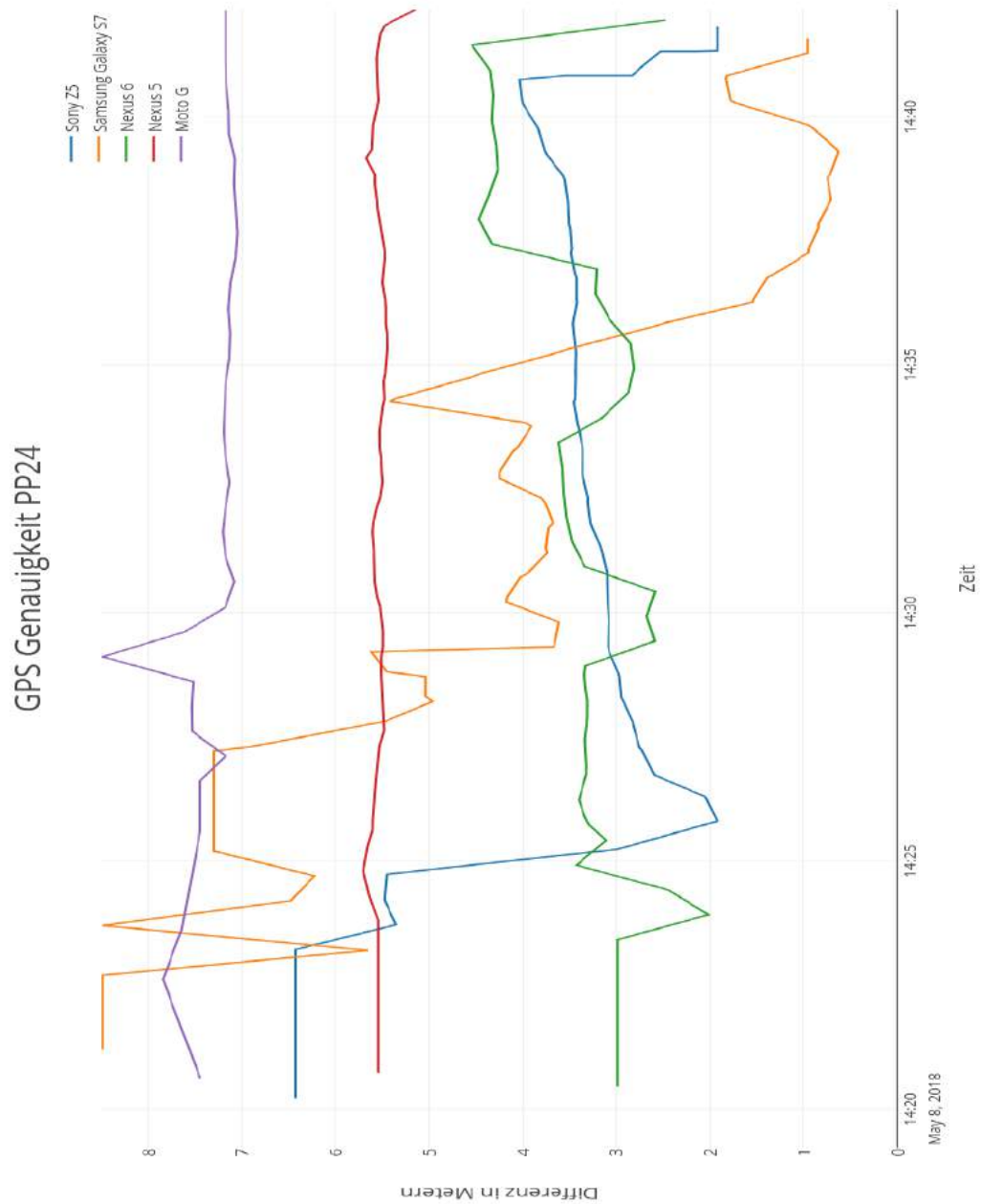


Abbildung 5.36: Die Genauigkeit der Smartphones am Referenzpunkt PP24 liegt zwischen 0.7 und 7 Metern. Dabei erlangt das Samsung Galaxy S7 mit einer Positionsgenauigkeit von ca. 0.7 Metern die besten und das Moto G die schlechtesten Werte von ca. 7 Metern.

## 5 Ergebnis

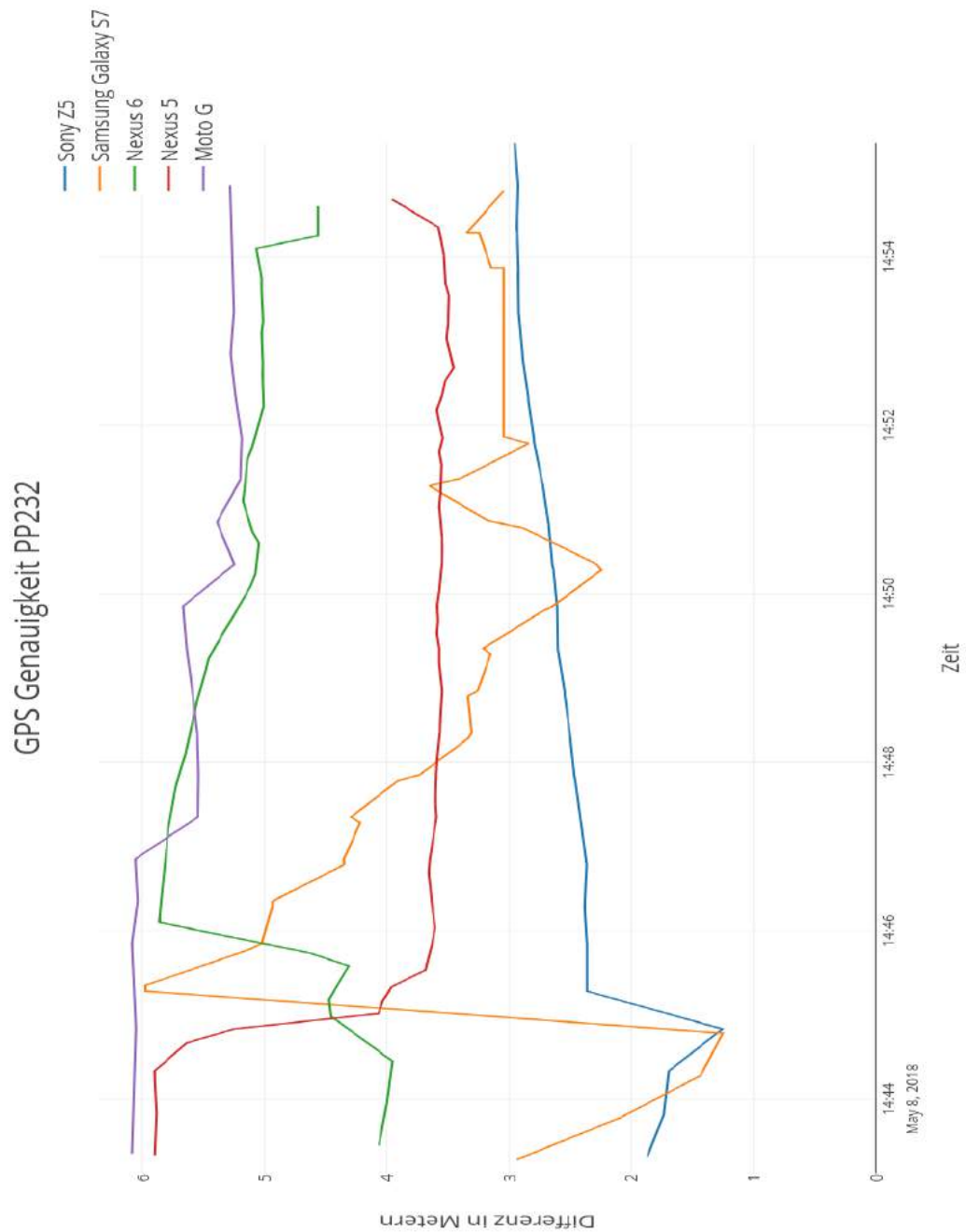


Abbildung 5.37: Die Genauigkeit der Smartphones am Referenzpunkt PP232 liegt zwischen 1.3 und 5.3 Metern. Dabei erlangt das Sony Z5 mit einer Positionsgenauigkeit von ca. 1.3 Metern die besten und das Moto G die schlechtesten Werte von ca. 5.3 Metern.

## 5 Ergebnis

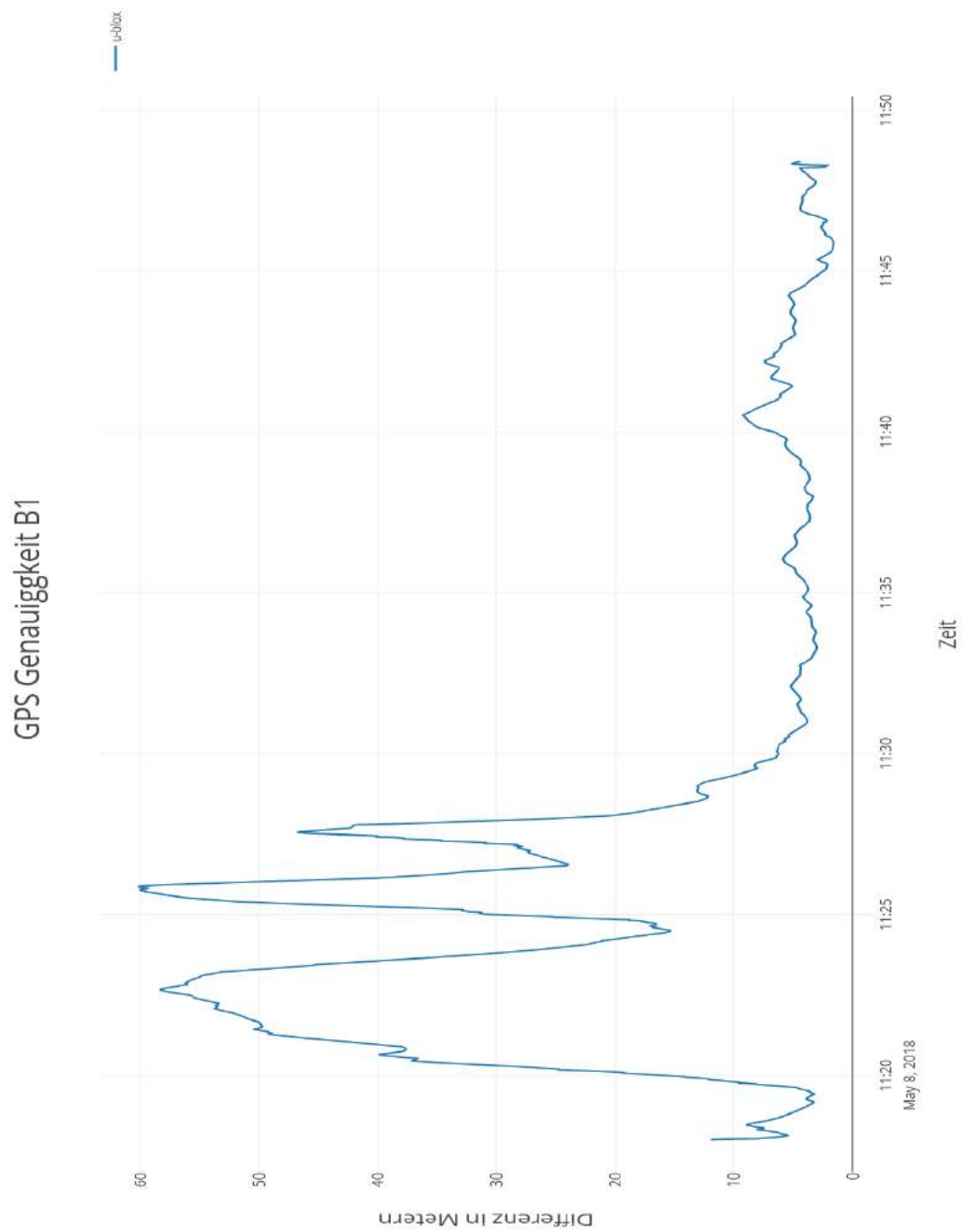


Abbildung 5.38: Die Positionsgenauigkeit des GPS Gerätes von u-Blox lag am Referenzpunkt B1 bei ca. 3 Metern.

## 5 Ergebnis

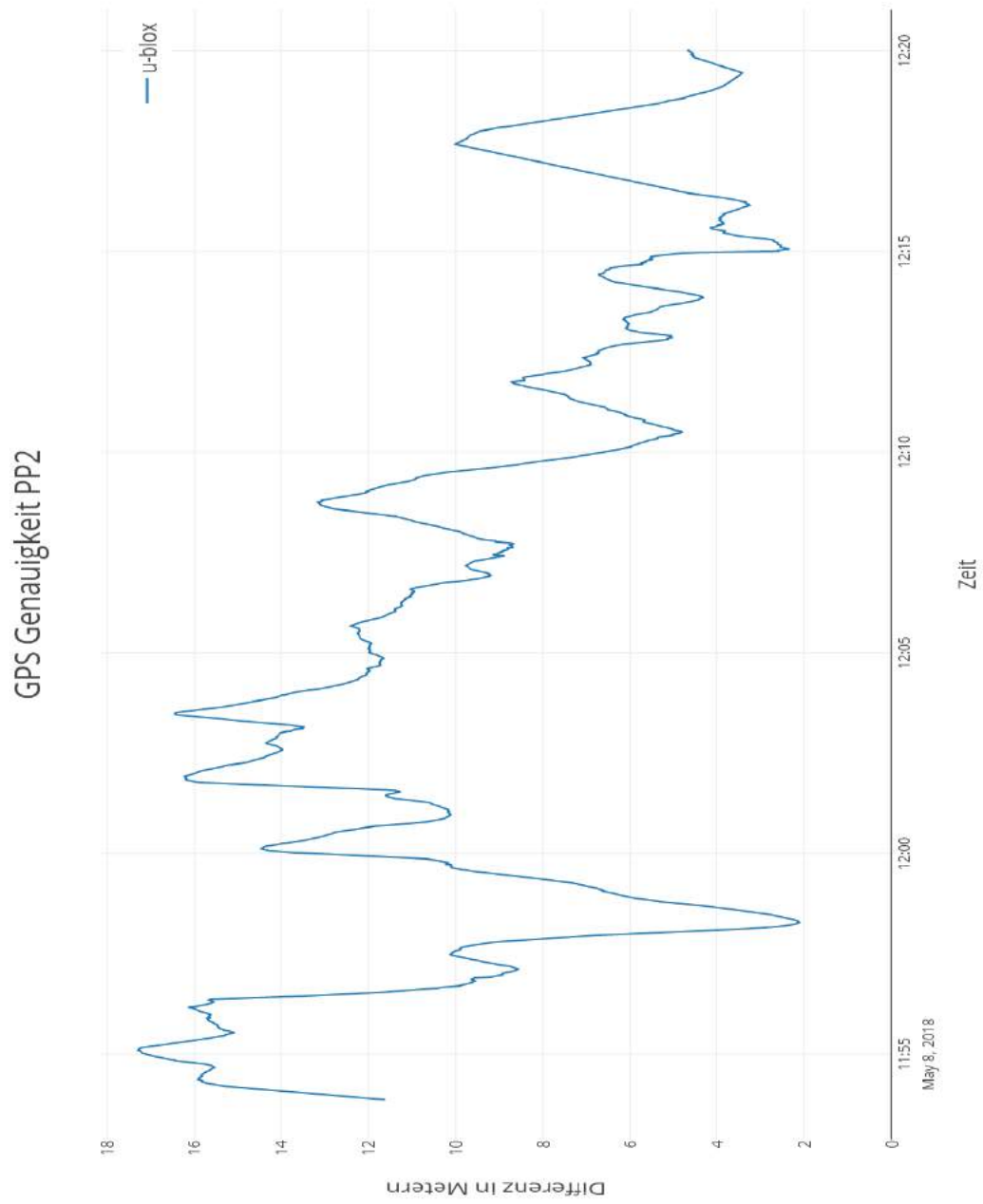


Abbildung 5.39: Die Positionsgenauigkeit des GPS Gerätes von u-Blox lag am Referenzpunkt PP2 bei ca. 2 Metern.

## 5 Ergebnis

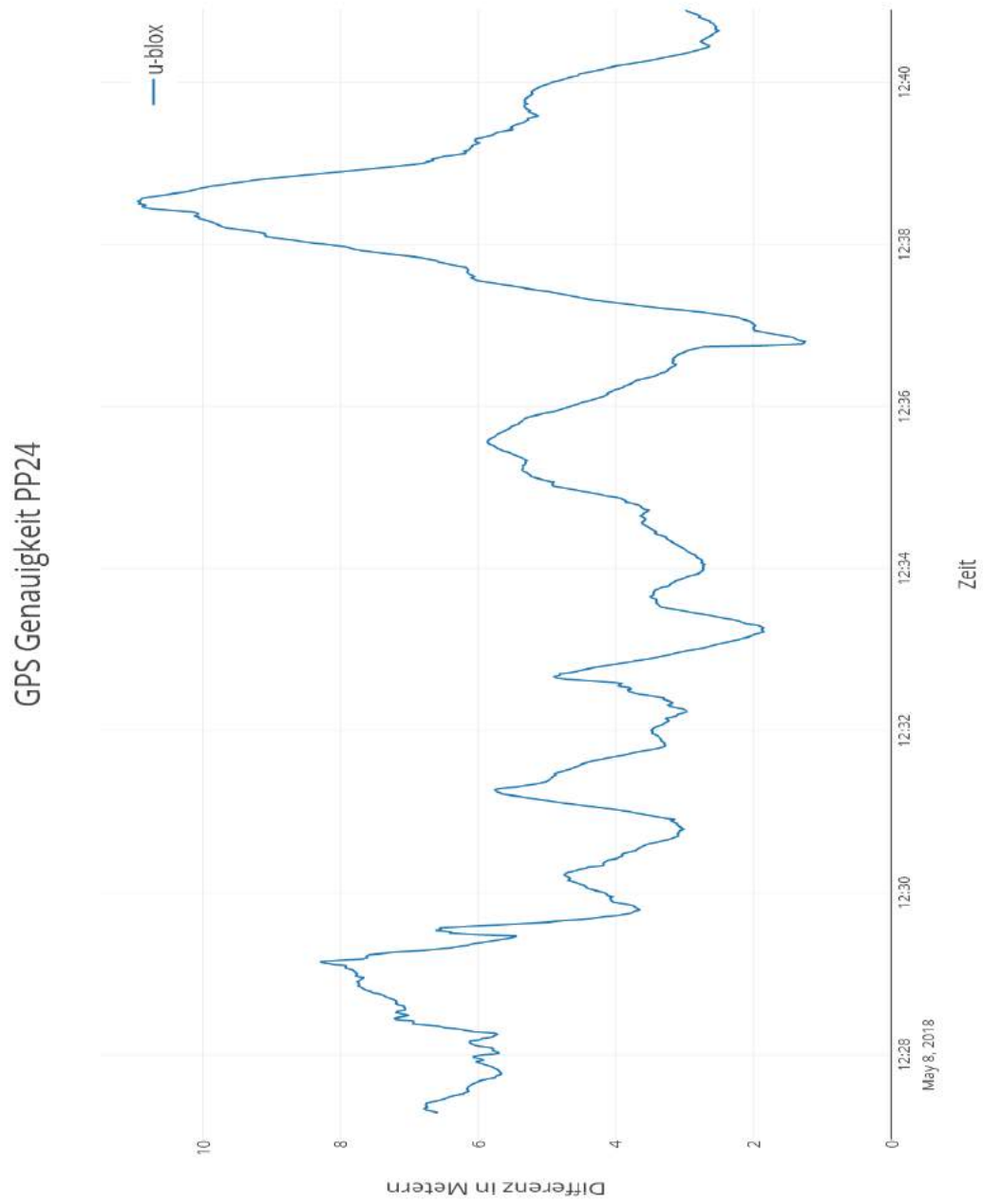


Abbildung 5.40: Die Positionsgenauigkeit des GPS Gerätes von u-Blox lag am Referenzpunkt PP24 bei ca. 1.7 Metern.

## 5 Ergebnis

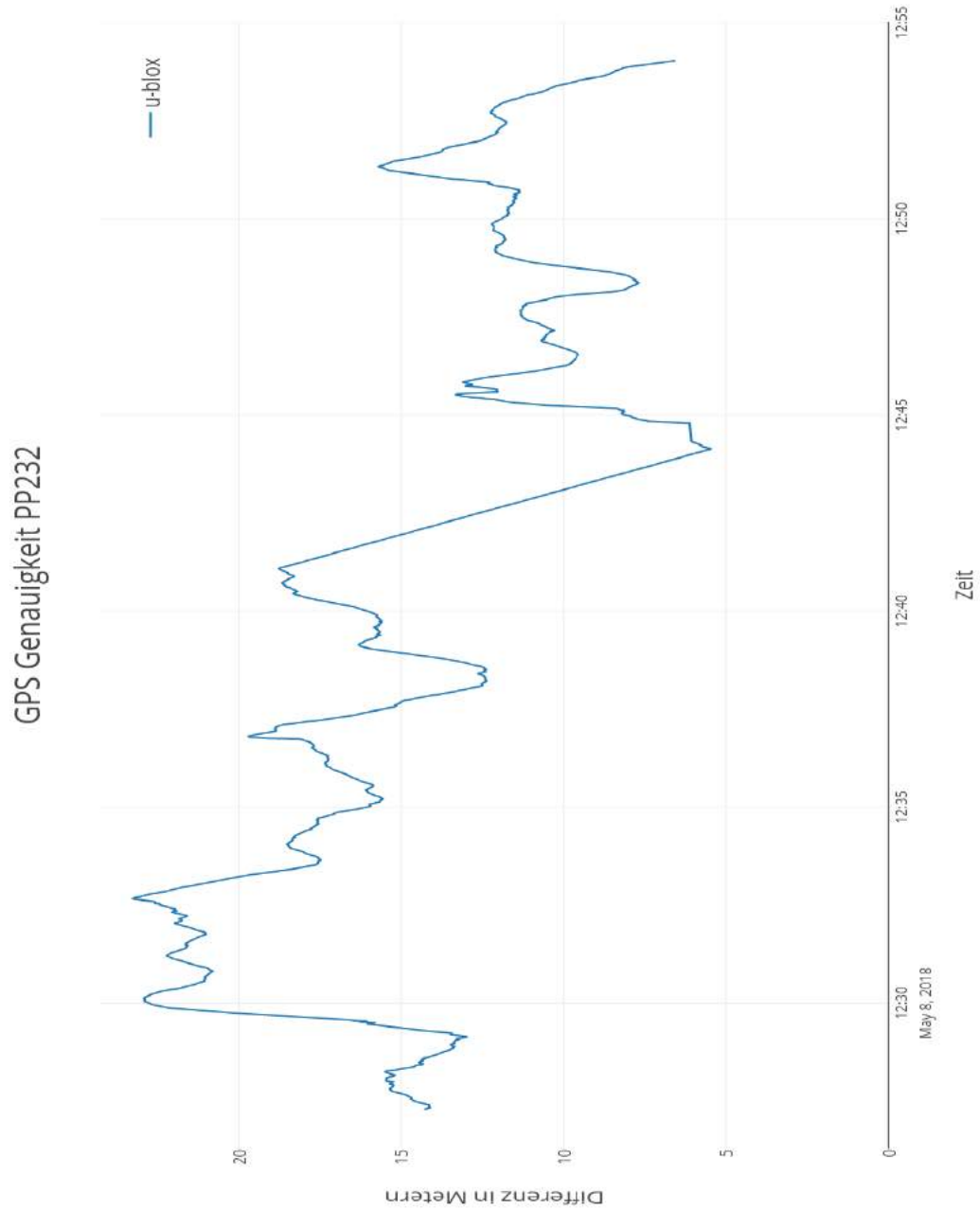


Abbildung 5.41: Die Positionsgenauigkeit des GPS Gerätes von u-Blox lag am Referenzpunkt PP232 bei ca. 5.2 Metern.

## 5 Ergebnis

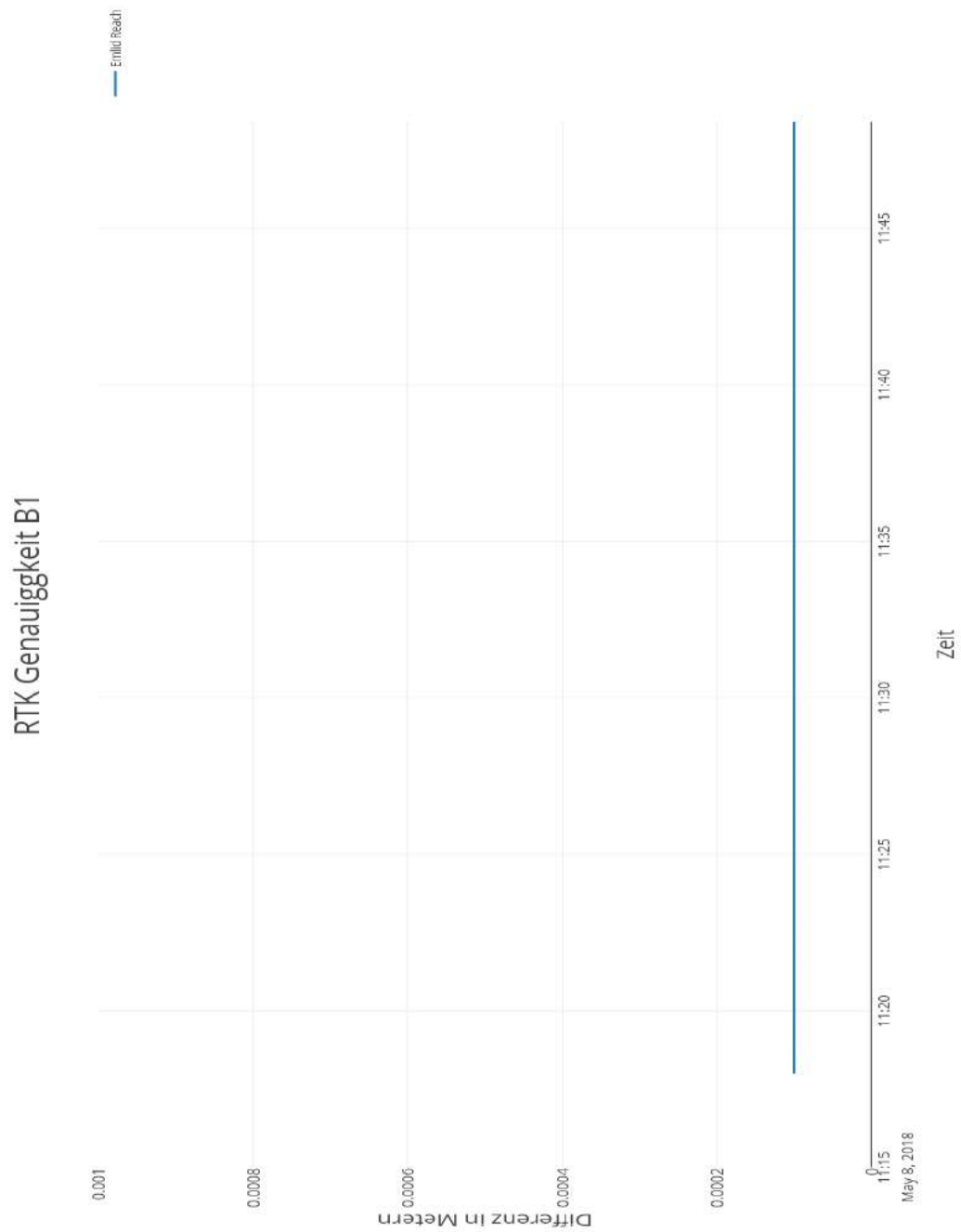


Abbildung 5.42: Die Positionsgenauigkeit des RTK Gerätes von Emlid lag am Referenzpunkt B1 bei weniger als einen Millimeter.

## 5 Ergebnis

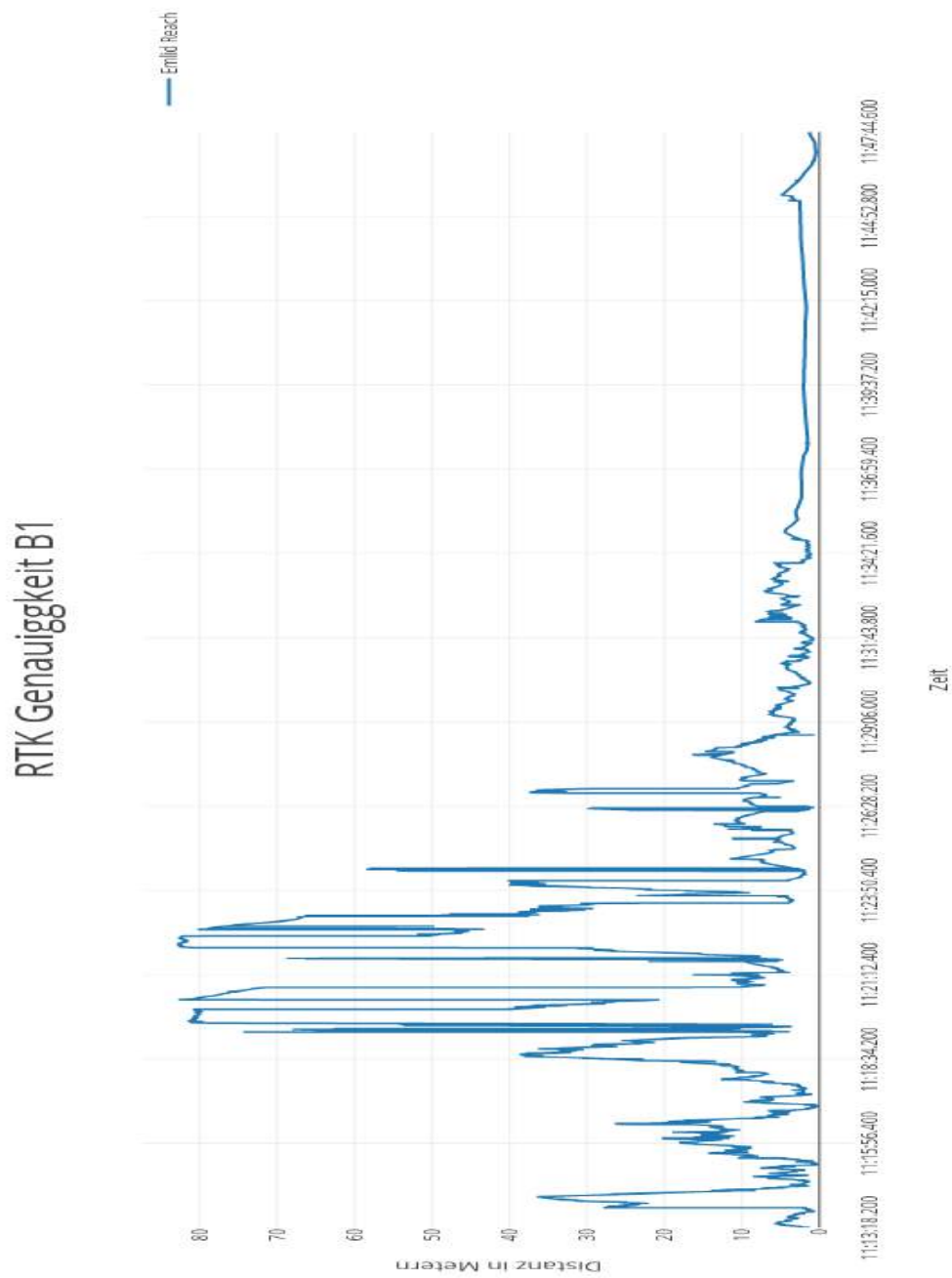


Abbildung 5.43: Die Positionsgenauigkeit der einzelnen RTK Roverstation von Emlid lag am Referenzpunkt B1 im besten Fall bei ca. 14 Zentimetern.



## 5 Ergebnis



Abbildung 5.44: Die Positionsgenauigkeit des RTK Gerätes von Emlid lag am Referenzpunkt PP2 bei ca. 1.8 Millimetern.

## 5 Ergebnis

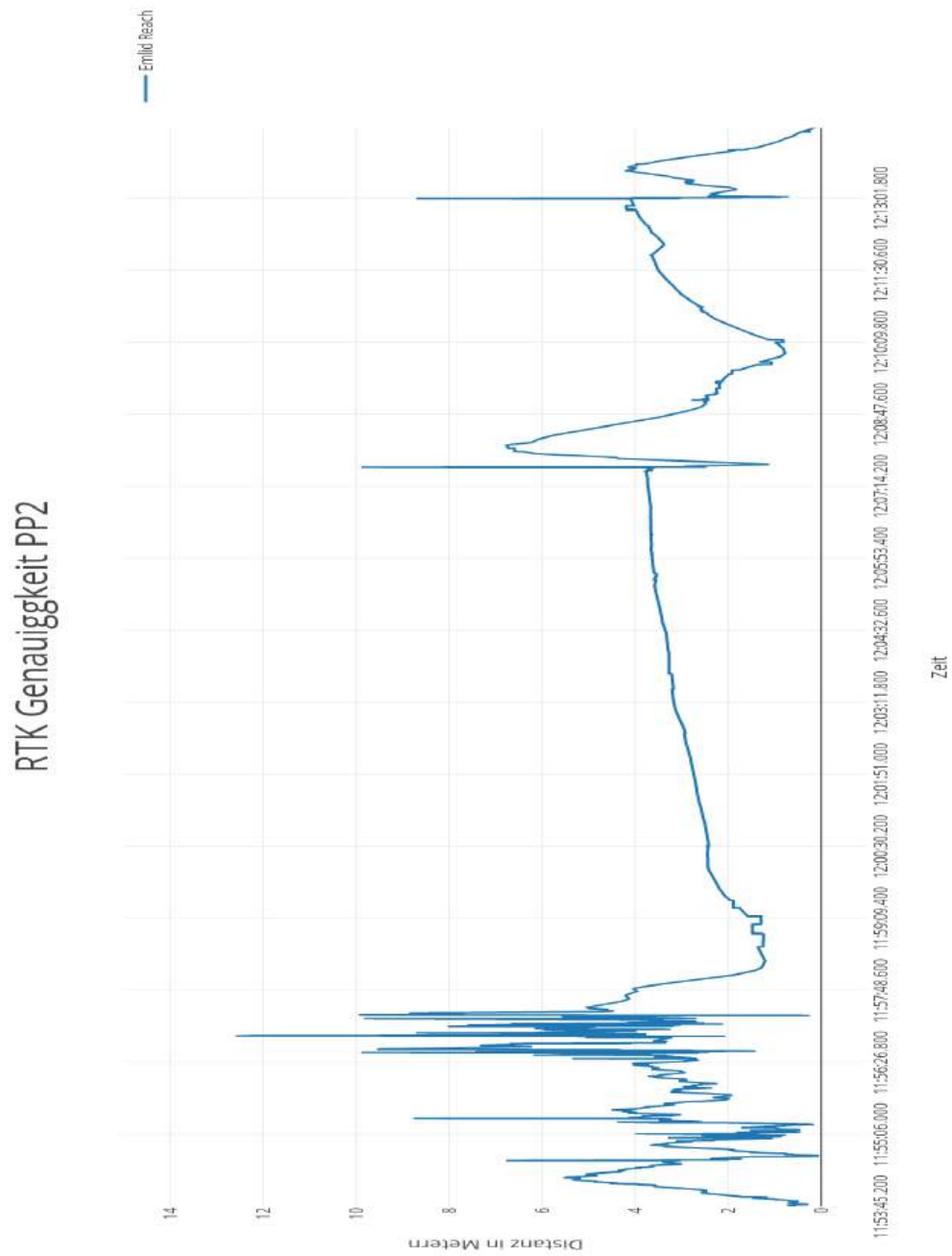


Abbildung 5.45: Die Positionsgenauigkeit der einzelnen RTK Roverstation von Emlid lag am Referenzpunkt PP2 im Besten Fall bei ca. 7 Zentimetern.

## 5 Ergebnis

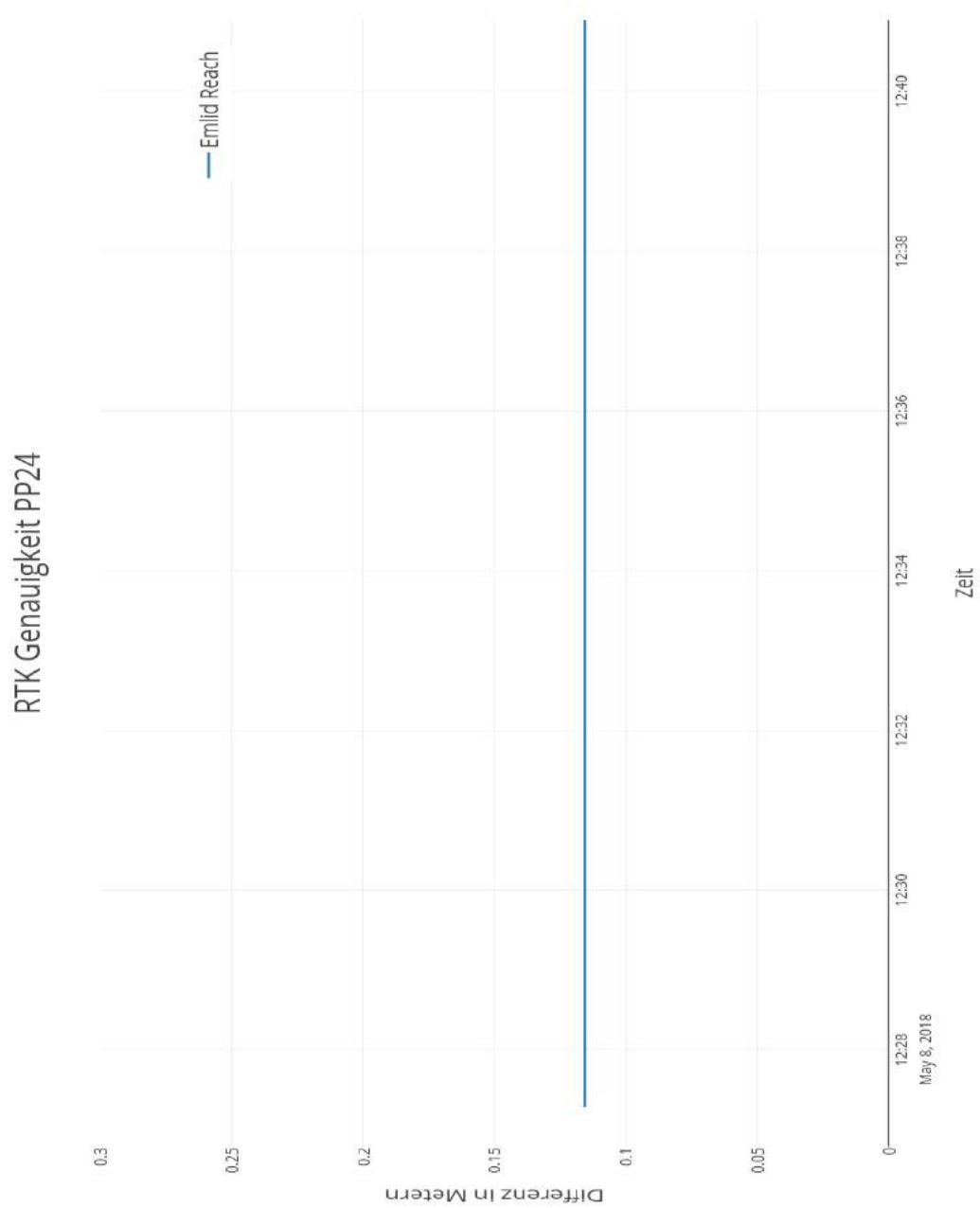


Abbildung 5.46: Die Positionsgenauigkeit des RTK Gerätes von Emlid lag am Referenzpunkt PP24 bei ca. 12 Zentimetern.

## 5 Ergebnis

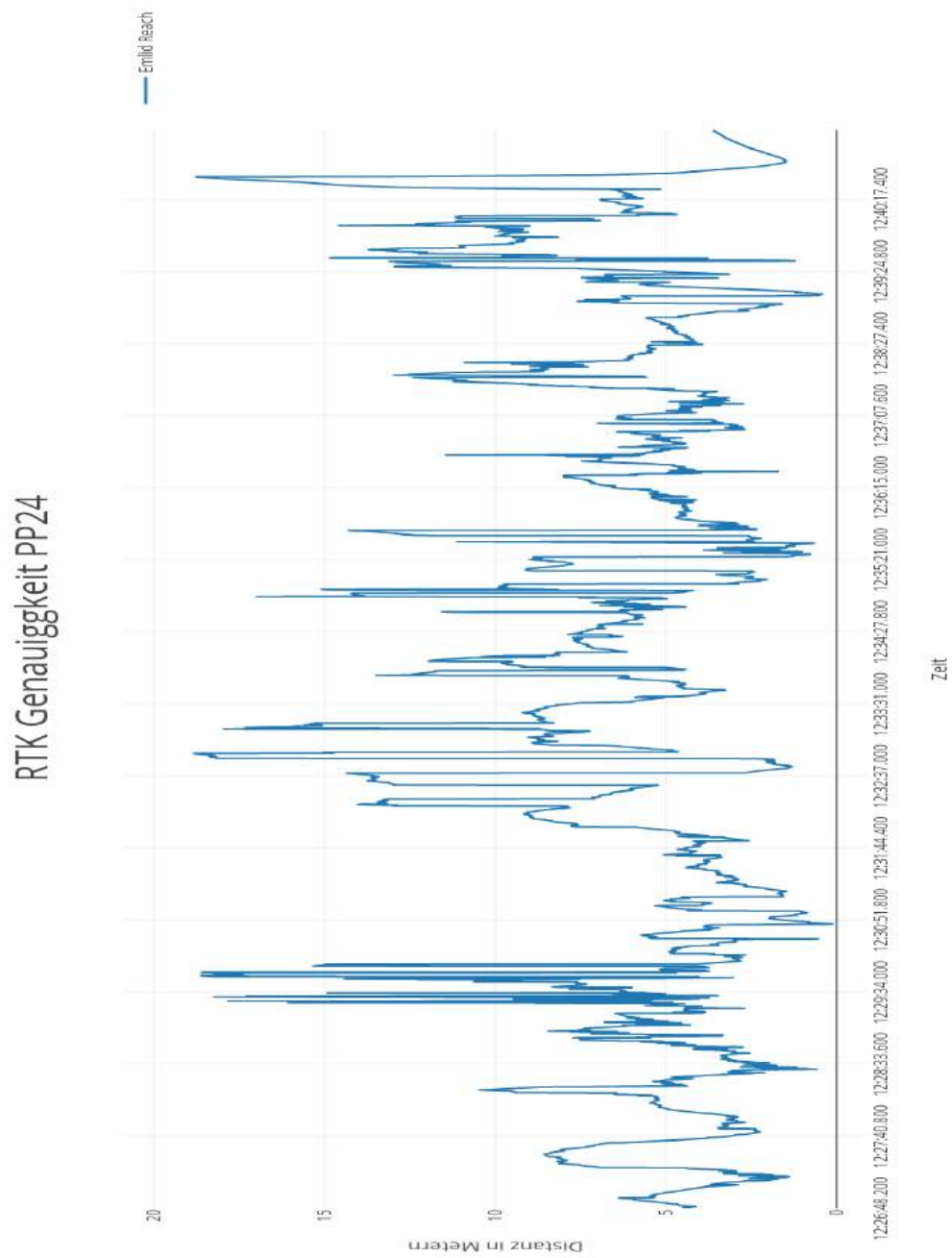


Abbildung 5.47: Die Positionsgenauigkeit der einzelnen RTK Roverstation von Emlid lag am Referenzpunkt PP24 im Besten Fall bei ca. 13 Zentimetern.

## 5 Ergebnis

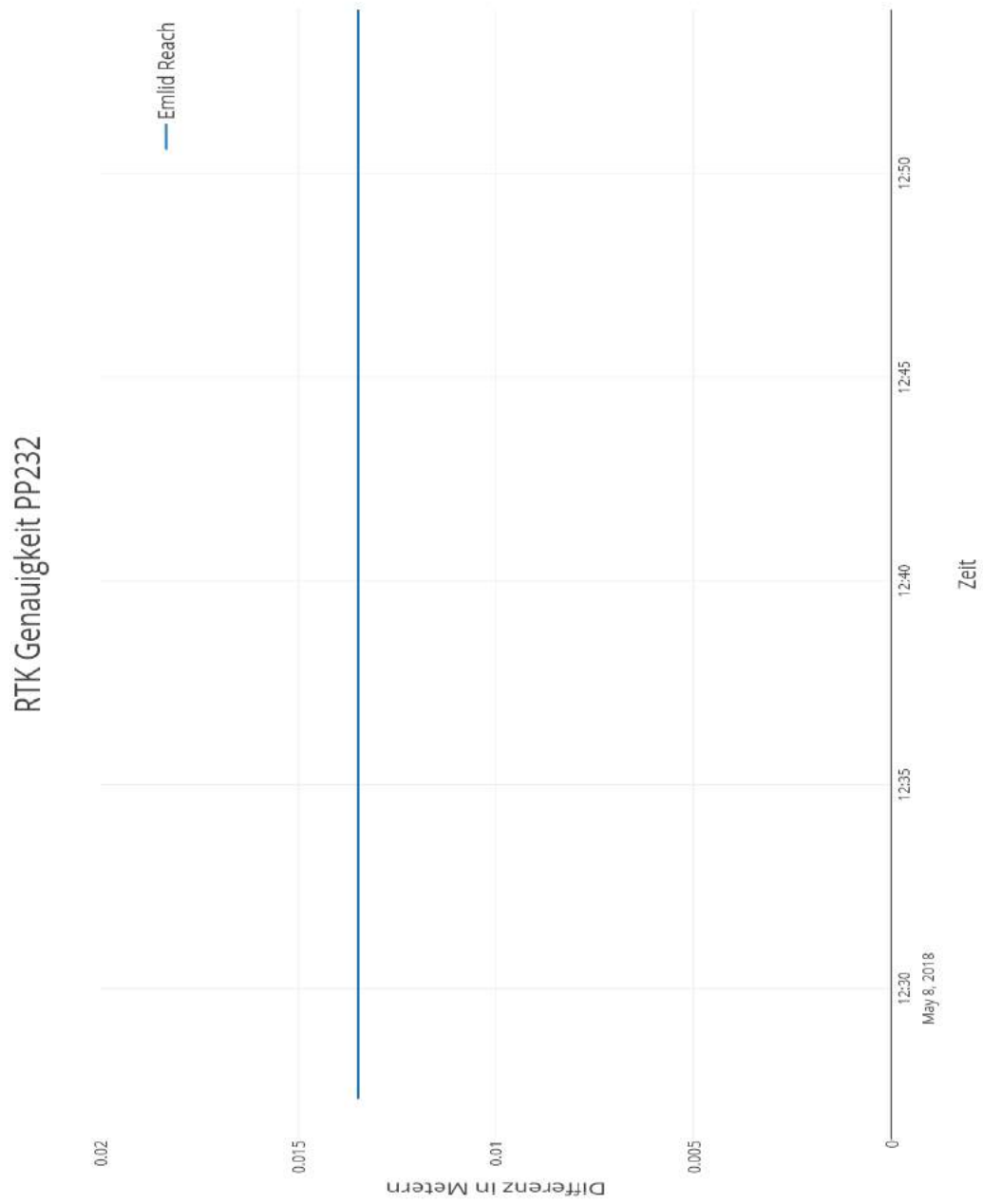


Abbildung 5.48: Die Positionsgenauigkeit des RTK Gerätes von Emlid lag am Referenzpunkt PP232 bei ca. 13 Millimetern.

## 5 Ergebnis

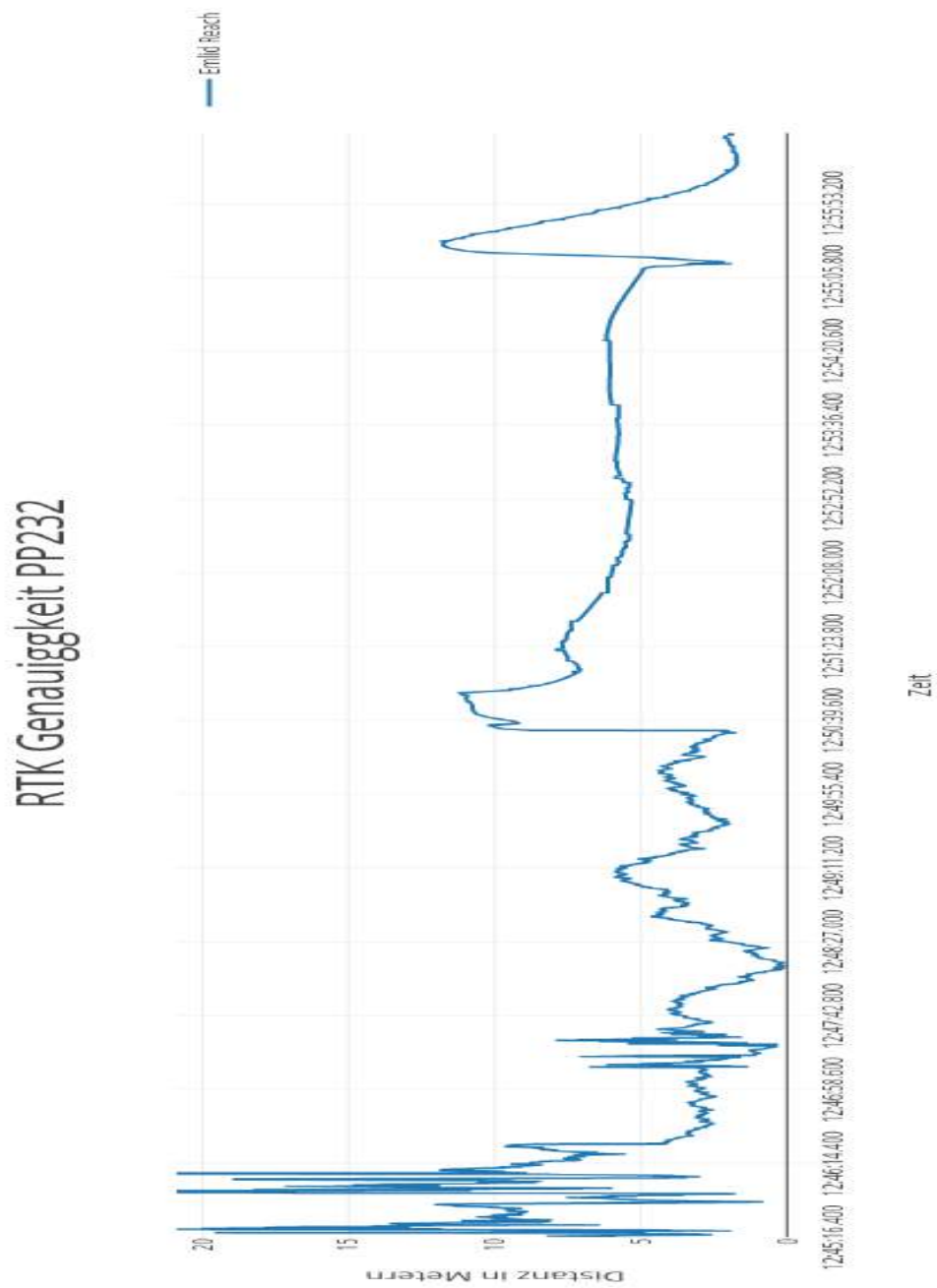


Abbildung 5.49: Die Positionsgenauigkeit der einzelnen RTK Roverstation von Emlid lag am Referenzpunkt PP232 im besten Fall bei ca. 6 Zentimetern.

## 5 Ergebnis

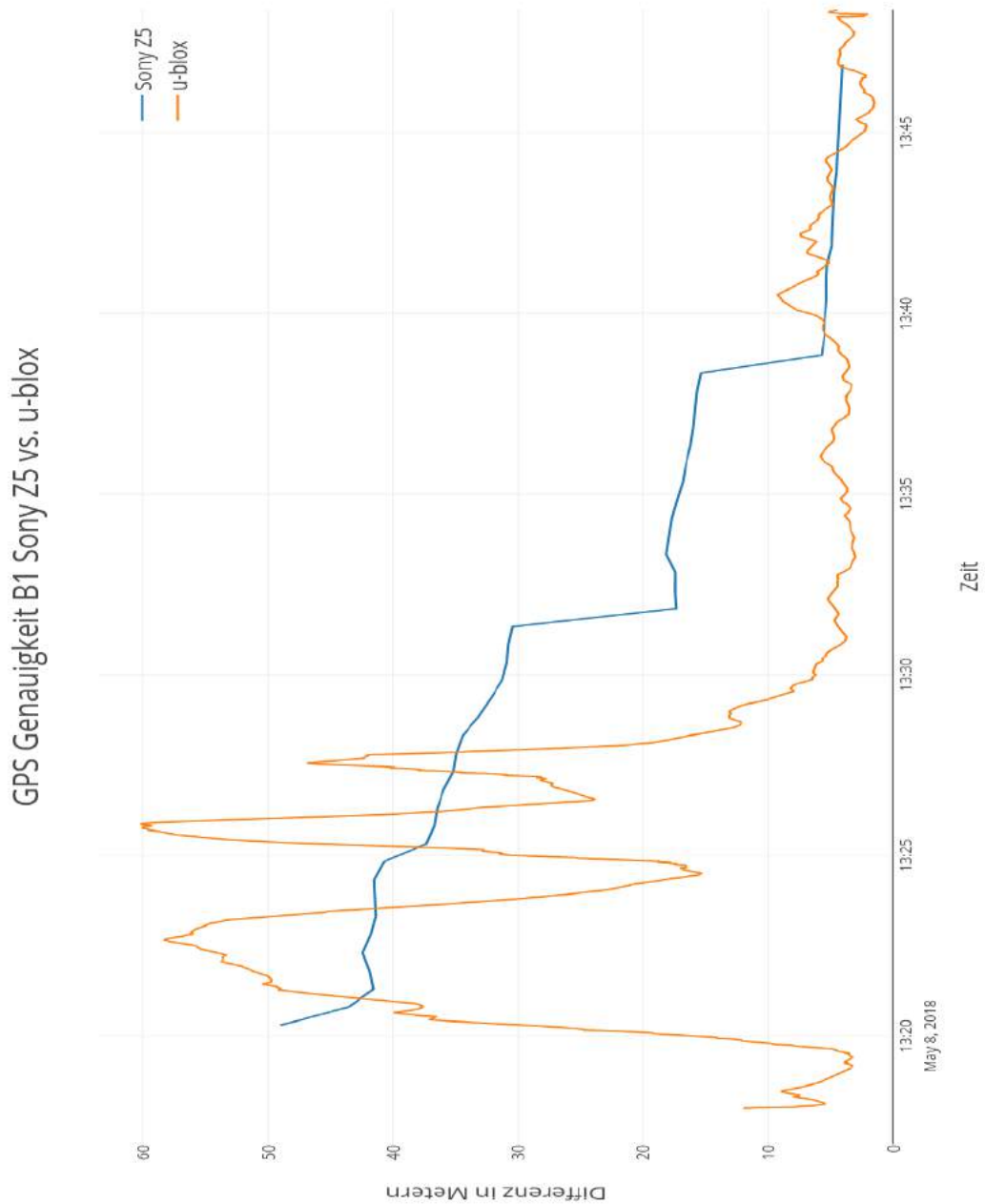


Abbildung 5.50: Am Referenzpunkt B1 sieht man, dass sich das u-Blox GPS System ca. 10 Minuten schneller eine gute Positionsgenauigkeit ermittelt, am Ende liegt die Genauigkeit jedoch sowohl beim Sony Z5, als auch beim u-Blox GPS System bei ca. 3-4 Metern.

## 5 Ergebnis

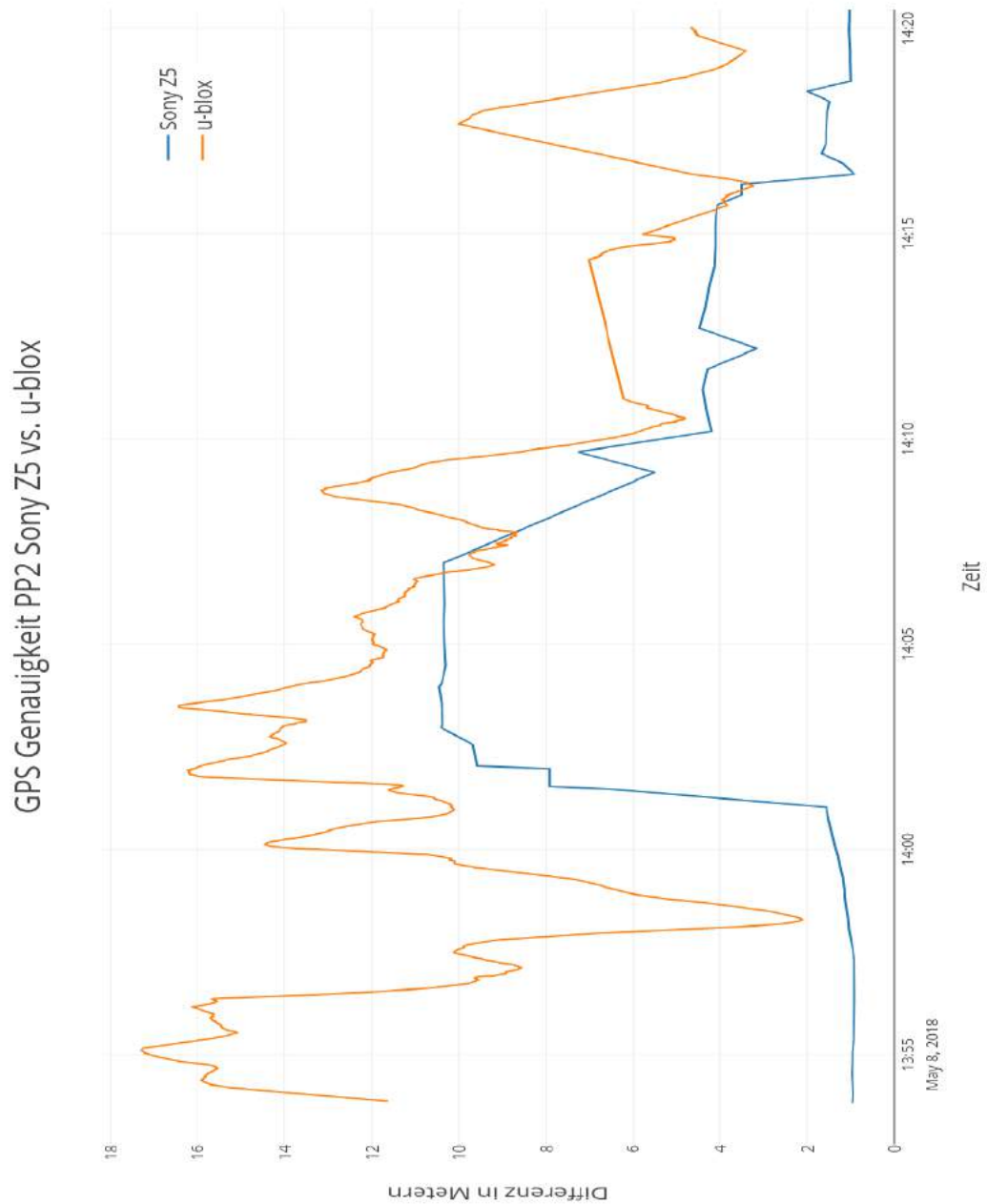


Abbildung 5.51: Am Referenzpunkt PP2 liegt die Genauigkeit beim Sony Z5 bei ca. 0.8 Metern, beim u-Blox GPS System bei ca. 3 Metern.



## 5 Ergebnis

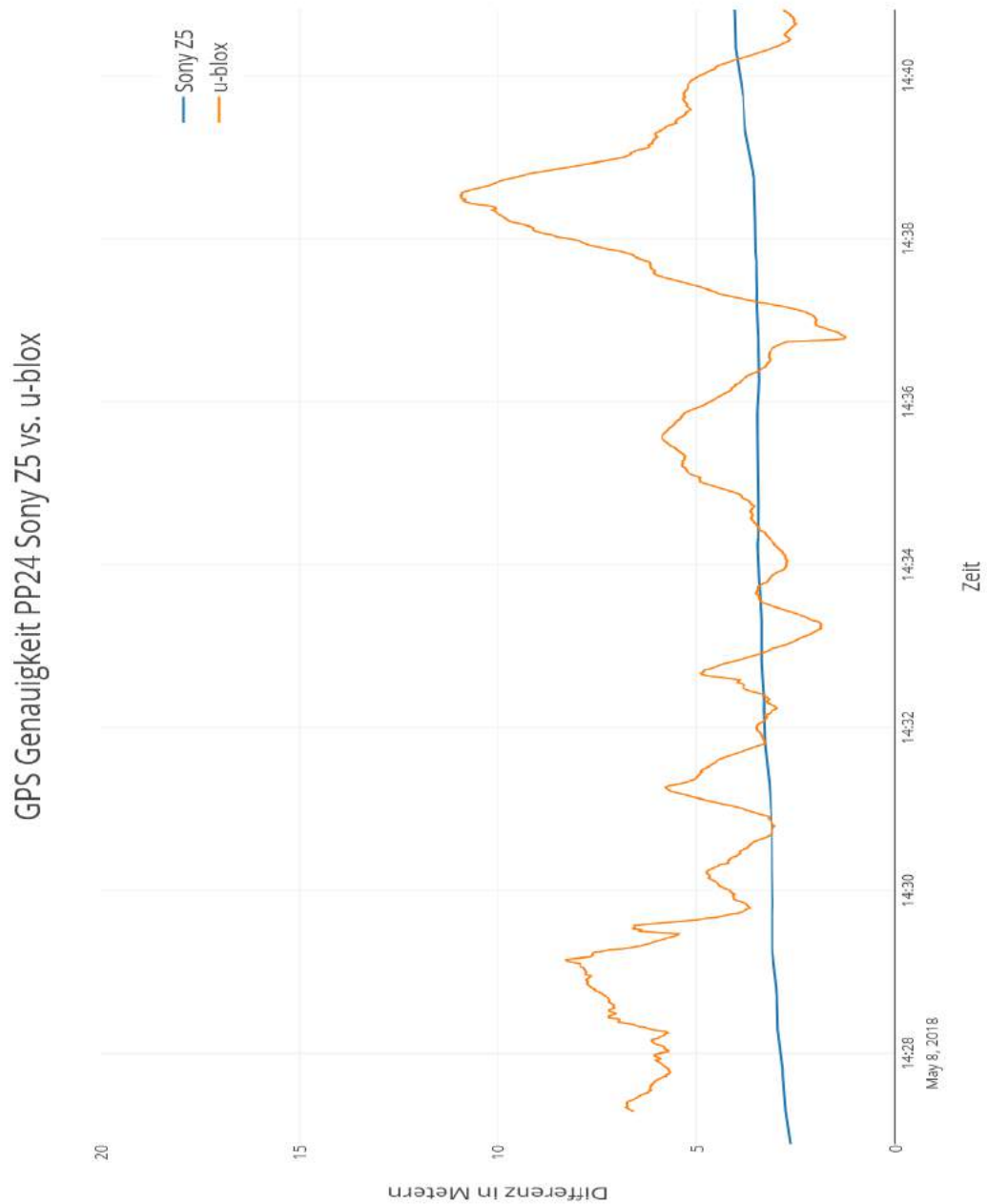


Abbildung 5.52: Am Referenzpunkt PP24 ist die Stabilität des GPS Empfanges beim Sony Z5 sehr gut. Die Positionsgenauigkeit beim Sony Z5 liegt bei ca. 3 Metern, beim u-Blox GPS System bei ca. 2 Metern.

## 5 Ergebnis

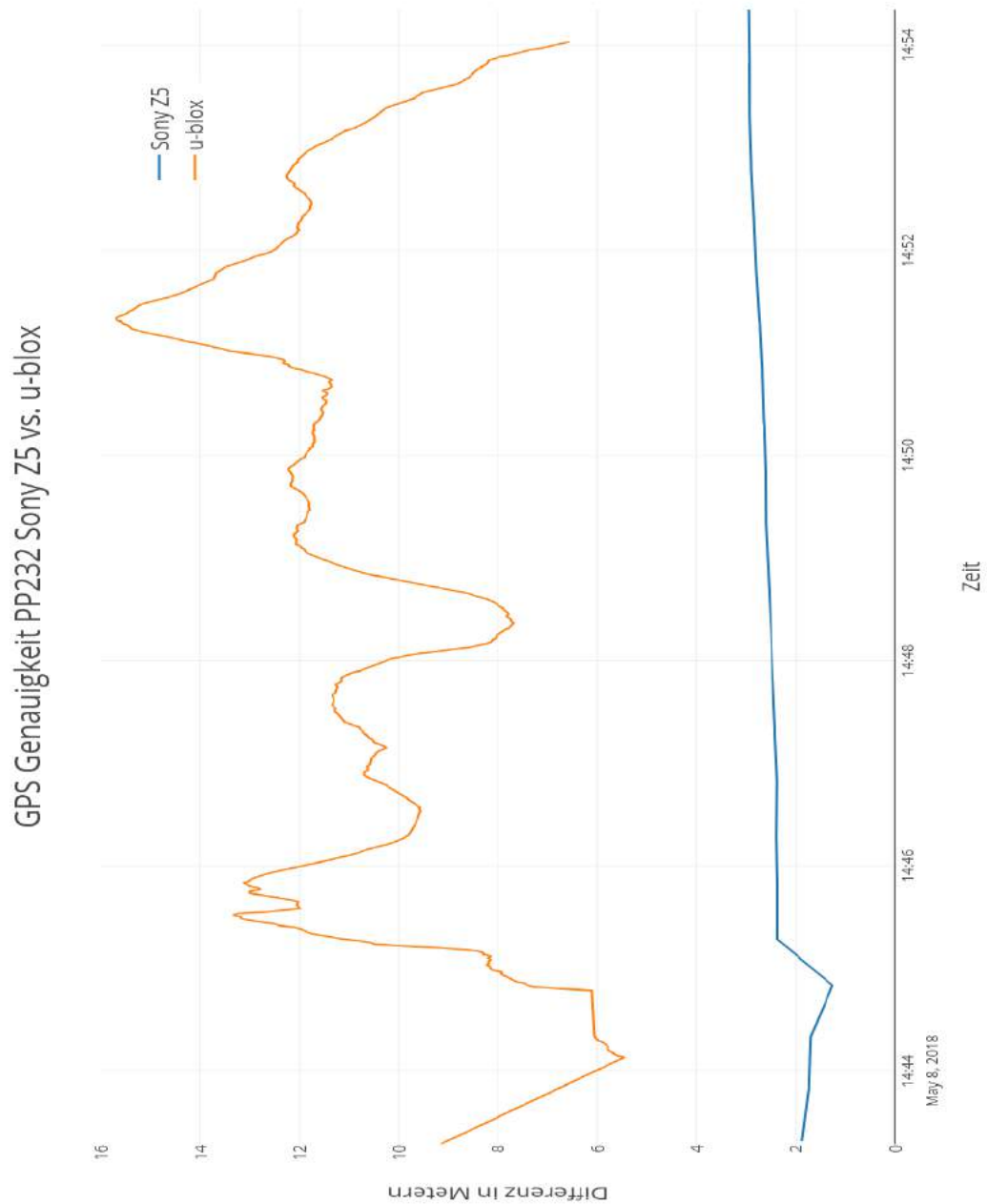


Abbildung 5.53: Am Referenzpunkt PP24 ist die Stabilität des GPS Empfanges beim Sony Z5 sehr gut. Die Positionsgenauigkeit beim Sony Z5 liegt bei ca. 2 Metern, beim u-Blox GPS System bei ca. 5 Metern.

## 5 Ergebnis

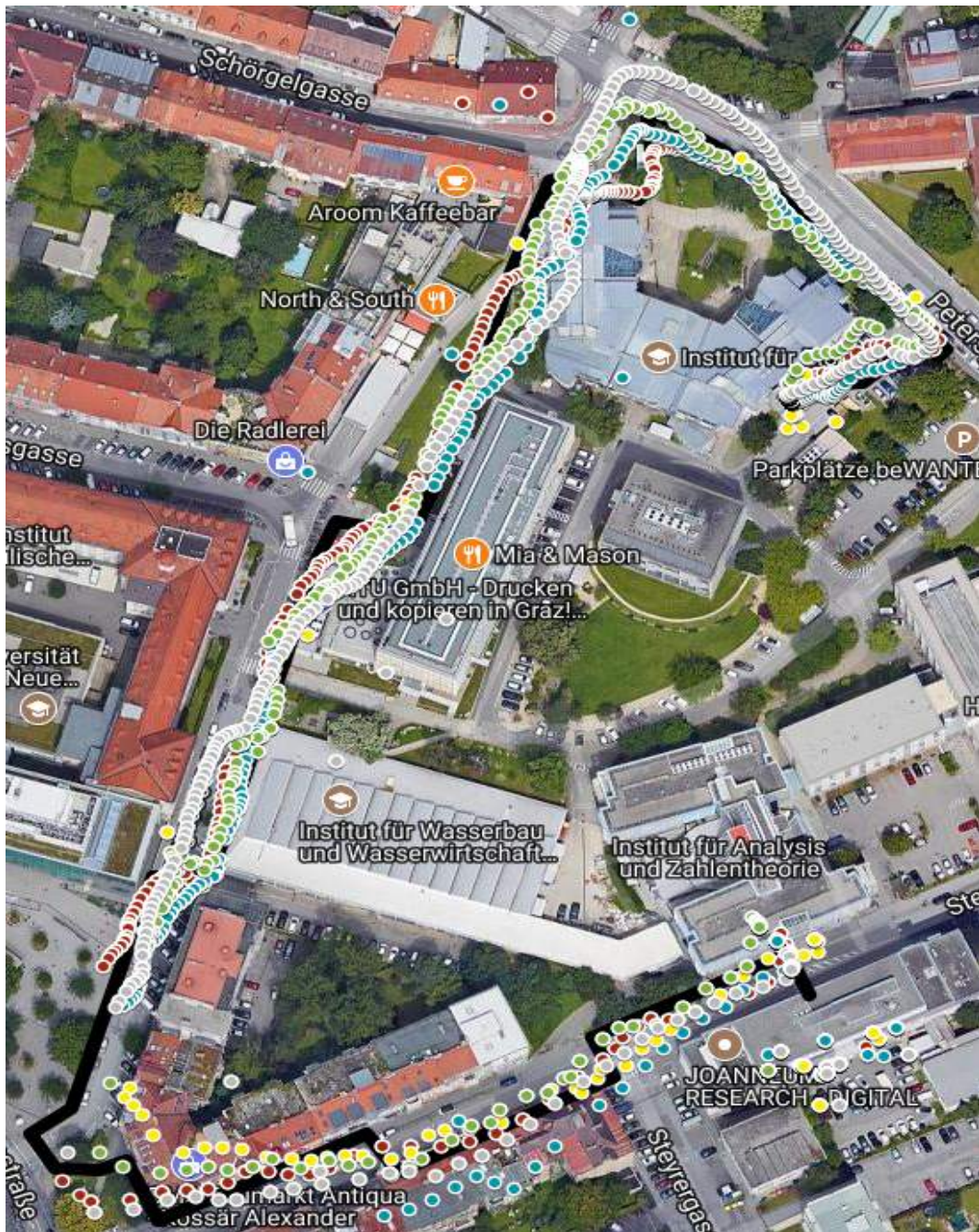


Abbildung 5.54: Wie in der Abbildung ersichtlich, erkennt man, dass sich alle GPS Sensoren innerhalb von wenigen Metern zu der abgelaufenen Route befinden.

## 6 Schlussfolgerung

Ziel dieser Masterarbeit war es ein Entwicklungskonzept für eine mobile Kommunikations- und Multisensorlösung für Sicherheits- und Risikomanagement im Freiland und im Objektschutz zu erarbeiten, testen und evaluieren. Hauptaugenmerk soll dabei das Monitoring im Rahmen von Sicherungsaufgaben bei urbanen Plätzen und Großveranstaltungen sein.

Der erste Schritt der Arbeit bestand darin, die diversen Sensoren, welches für das Mastsystem verwendet werden sollen, zu evaluieren. Das System soll einerseits aus Kameras bestehen, die die aktuelle Lage einerseits als Fotos, aber auch als Videos aufnehmen sollten. Gleichzeitig ist auch eine 360 Grad Kamera angedacht um sehr schnell, das komplette Gelände überwachen zu können. Da die Fotos, als auch die Videos als Livestream an die Zentrale gesendet werden sollten, wird auch zusätzlich Hardware benötigt, die diese Aufgaben erledigen sollen.

Außerdem wird eine Positionsbestimmung, als auch eine Richtungsbestimmung benötigt. Die Kamera sollte auch im fahrenden Betrieb wackelfreie Bilder und Videos liefern, daher wird auch noch eine Stabilisierungshardware benötigt, die dies ermöglichen soll.

Nach einer ersten ausführlichen Recherche, wurde der erste Prototyp mit den ausgewählten Sensoren entwickelt. Dieser Prototyp konnte danach am Frequency Festival in St. Pölten erstmals ausgetestet werden.

Die Eindrücke und Ergebnisse aus diesem Test wurden ausgewertet und daraufhin wurde eine zweite Rechercherunde gestartet, um die Probleme, die bei dem ersten Prototyp auftauchten, lösen zu können. Nach der

## 6 Schlussfolgerung

ausführlichen Recherche der Sensoren, wurde ein zweiter Prototyp entwickelt, getestet und daraufhin evaluiert.

Zusätzlich wurden Global Positioning System (GPS)- und Inertial Measurement Unit (IMU) Genauigkeitstests durchgeführt, um für das fertige Mastsystem, das optimale Geräte zu finden, welches einerseits eine bestimmte Genauigkeit liefert, aber auch kostengünstig und platzsparend ist.

Da eine kompakte, kostengünstige, aber auch innovative und leistungsstarke Endlösung entwickelt werden sollte, wurde nach Auswertung aller Daten diese verschiedenen Komponenten für das Endprodukt ausgewählt.

Für die Fotos, sowie den Livestream wurde das Sony Z5 Smartphone ausgewählt, da es neben einer 23 Megapixel Kamera, auch eine unkomplizierte Möglichkeit bietet, die Bilder und den Livestream weiterzuleiten. Da sich jedoch bei den Tests mit dem ersten Prototypen am Frequency Festival herausstellte, dass die Stabilisierungshardware Osmo DJI, bei den starken Unebenheiten mit dem Mastsystem am Quad nicht umgehen konnte und somit das Handy nicht stabilisieren konnte. Dadurch entschieden wir uns für die teurere Stabilisierungshardware, die Osmo Zenmuse X5R, bei der direkt eine Kamera eingebaut war und somit war diese auch austariert und führte auch bei starken Unebenheiten zu keinen Problemen. Aufgrund des teureren Preises, im Gegensatz zum Osmo DJI System, und der komplexeren Anbindung für die Bilder, sowie den Livestream, wäre eine weitere Recherche ratsam, um eine Stabilisierungshardware zu finden, die auch bei starken Unebenheiten ein Smartphone stabilisieren kann.

Neben einer normalen Kamera, wurde auch noch eine 360 Grad Kamera benötigt. Nach einer Recherche entschieden wir uns für die Ricoh Theta R, die neben der Full HD Auflösung, auch einen HD Livestream anbietet.

Damit es auch möglich ist die Livestream Daten der Smartphonekamera, als auch der der 360 Grad Kamera auf den Server zu streamen, wurde noch eine zusätzliche Hardware benötigt, die diese Aufgabe übernehmen soll. Dabei haben wir uns für den ODROID entschieden, da er leistungsstärker als der Raspberry PI war und zusätzlich eine eigene Kühlung besitzt.

## 6 Schlussfolgerung

Für die Positionsbestimmung, sowie der Richtungsbestimmung, wurde Tests mit verschiedenen Geräten durchgeführt. Bei den GPS Tests wurden die Genauigkeit verschiedener Geräte, für bestimmten Referenzpunkte getestet. Die Auswahl des Ortes für die Positionstests fiel auf das Gelände der Technischen Universität rund um die Steyrergasse, da dort vom Institut für Geodäsie ausgewählte Punkte genau vermessen wurden. Aus diesem Grund konnten genaue Referenzpunkte ermittelt werden, weshalb auch die Ergebnisse der Tests genauer ausfallen sollten.

Es wurde versucht verschiedene Punkte auszuwählen, einerseits Punkte die großteils frei von hohen Gebäuden sind, um auch einen besseren GPS Empfang zu erhalten und andererseits Punkte, die rundherum großteils von Gebäuden umgeben sind, um auch Referenzwerte für Einsätze in Großstädten zu erhalten.

Das Endresultat dieser GPS Tests zeigte, dass die Genauigkeit von einem RTK System, obwohl es eine low-cost Variante ist, mit einer Genauigkeit von wenigen Millimetern bzw. Zentimetern ganz klar die besten Werte aller getesteten GPS Sensoren erreichte.

Bei dem Vergleich zwischen dem professionellen GPS System und den Handy GPS Sensoren erkennt man, dass ältere Smartphones wie das Moto G nicht mit der Genauigkeit eines professionellen GPS Systemes mithalten können.

Wenn aber die neuen Smartphones, wie das Sony Z5, als auch das Samsung Galaxy S7 für die Tests hergenommen werden, sieht man, dass die GPS Sensoren dieser Smartphones gleich gute, teilweise sogar bessere Werte liefern, als das professionelle GPS System von u-blox.

Bei den IMU Genauigkeitstests wurde ein professioneller IMU Sensor, nämlich der LPMS-B von der Firma LP-Research verwendet. Nebenbei wurde noch das Sony Z5 und Samsung Galaxy S7 Smartphone benutzt und zusätzlich noch das RTK System von Emlid Reach.

Dabei wurden die Geräte mehrmals um eine bestimmte Gradanzahl im Kreis bewegt und dabei die yaw, pitch, sowie roll Werte entnommen. Diese Werte

## 6 Schlussfolgerung

wurden mit den tatsächlichen Daten verglichen und aus allen Ergebnissen für jedes Gerät ein Durchschnittsgenauigkeitswert in Grad ermittelt.

Dabei erkennt man, dass der professionelle IMU Sensor LPMS-B die genauesten Werte liefert, dicht verfolgt von der IMU aus dem RTK System. Die zwei Smartphones hatten eine Genauigkeit von ungefähr einem Grad und lagen aus dem Grund hinter den beiden anderen getesteten IMU Sensoren.

Da unsere Endlösung eine kompakte, kostengünstige, aber auch innovative und leistungsstarke Endlösung sein sollte, entschieden wir uns für das Sony Z5, welches eine ausreichende Genauigkeit für die Positions- und Richtungsbestimmung bietet und zusätzlich auch eine gute, einfache Anbindung ermöglicht.

Schlussendlich wurde eine fertige mobile Multisensorlösung zur Unterstützung von Sicherheits- und Risikoaufgaben bei Großveranstaltungen entwickelt, die Sicherheitsmitarbeiter bei Sicherungsaufgaben bei urbanen Plätzen und Großveranstaltungen unterstützen sollen und dadurch womögliche dramatische Tragödien verhindern können.

# Appendix



# Literatur

- [Ahm+13] N. Ahmad u. a. »Reviews on various inertial measurement unit (IMU) sensor applications«. In: *Int. J. Signal Process. Syst* (2013) (siehe S. 18, 20).
- [Bau13] C. Bauer. »On the (In-)Accuracy of GPS Measures of Smartphones: A Study of Running Tracking Applications«. In: *Proceedings of International Conference on Advances in Mobile Computing & Multimedia* (2013) (siehe S. 24, 27).
- [Cik+16] Michael Cik u. a. *AGETOR-Analyse von Bewegungsströmen von Personen in Echtzeit auf Basis von Daten aus Mobilfunk und sozialen Medien zur Gewährleistung der Sicherheit bei Großveranstaltungen im urbanen und nicht urbanen Raum: Wissenschaftlicher Endbericht*. FTI-Initiative: KIRAS. ., März 2016 (siehe S. 22).
- [FER10] S. Frings, W. Engelbach und D. L. Remondes. »Informations- und Kooperationsportal als Unterstützungssystem der Behörden und Organisationen mit Sicherheitsaufgaben für Großveranstaltungen«. In: *Fraunhofer Institut für Arbeitswirtschaft und Organisation (IAO)* (2010) (siehe S. 23).
- [Gar11] Marko Gargenta. *Einführung in die Android-Entwicklung*. 2011. ISBN: 978-3868991147 (siehe S. 11).
- [Geio5] Gebhard Geiger. »Europas weltraumgestützte Sicherheit - Aufgaben und Probleme der Satellitensysteme Galileo und GMES. SWP-Studie«. In: *Deutsches Institut für Internationale Politik und Sicherheit* (2005) (siehe S. 16).

## Literatur

- [Gre+06] D. A. Grejner-Brzezinska u. a. »Multi-sensor personal navigator supported by human motion dynamics model«. In: *Proceeding of 3rd IAG Symposium on Geodesy for Geotechnical and Structural Engineering and 12th FIG Deformation Measurement Symposium* (Mai 2006) (siehe S. 25, 29).
- [HLWo8] B Hofmann-Wellenhof, H Lichtenegger und E Wasle. »GNSS-Global Navigation Satellite Systems: GPS, GLONASS, Galileo & More«. In: *Springer-Verlag Wien* 518 (2008). DOI: 10.1007/978-3-211-73017-1 (siehe S. 16).
- [Men+11] T. Menard u. a. »Comparing the GPS capabilities of the Samsung Galaxy S, Motorola Droid X, and the Apple iPhone for vehicle tracking using FreeSim Mobile«. In: *In Proceedings of the 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)* (2011). DOI: 10.1109/ITSC.2011.6083141 (siehe S. 24, 26).
- [Pio13] H. F. Piotraschke. »RTK für Arme - Hochpräzise GNSS-Anwendungen mit den kostengünstigsten Trägerphasen-Rohdatenempfängern«. In: *Jahrestagung der Gesellschaft für Informatik in der Land-, Forst- und Ernährungswirtschaft e.V.* (2013) (siehe S. 18).
- [Roß+08] H. Roßnagel u. a. »Mobile Dienste zur Erhöhung der Sicherheit bei Großveranstaltungen«. In: *Fraunhofer IRB Verlag* (2008) (siehe S. 23).
- [SK13] J. Stähli und S. Keller. »Präzise Positionsbestimmung mit Low-Cost-GPS und Postprocessing.« In: (2013) (siehe S. 17).
- [Ven14] M. Venselaar. »Towards location- and orientation-aware gaming: Research on Location-based Games with additional compass features«. In: *Utrecht University* (2014) (siehe S. 24, 25, 28).
- [Vie10] S Vieweg. »Microblogged contributions to the emergency arena: Discovery, interpretation and implications«. In: *Computer Supported Collaborative Work* (2010) (siehe S. 22).
- [Wen17] G. Wendl. »Ackerbau – technische Lösungen für die Zukunft«. In: *Bayerische Landesanstalt für Landwirtschaft (LfL)* (2017) (siehe S. 18).

## Literatur

- [Zac14] Christian Zacherle. *Crowd Management: Verhinderung von Massenphänomenen bei Großveranstaltungen*. 2014. ISBN: 978-3-8428-9621-5 (siehe S. 1).
- [Zhe+16] Xu Zheng u. a. »Crowdsourcing based description of urban emergency events using social media big data«. In: *IEEE Transactions on Cloud Computing* (2016). DOI: 10.1109/TCC.2016.2517638 (siehe S. 22).