

# Schlussbericht

---

zu IGF-Vorhaben Nr. 19572 BR

## Thema

Entwicklung eines universellen Mess- und Auswertungssystems zur zeitlich begrenzten Analyse logistischer Systeme (TALOGS)

## Berichtszeitraum

01.06.2017 - 31.01.2020

## Forschungsvereinigung

Bundesvereinigung Logistik (BVL) e. V.

## Forschungseinrichtung(en)

TU Dresden - Professur für Technische Logistik (TL)

Dresden, 31.01.2020

---

Ort, Datum



Prof. Dr.-Ing. habil. Thorsten Schmidt

---

Name und Unterschrift aller Projektleiterinnen und Projektleiter der  
Forschungseinrichtung(en)

Gefördert durch:



Bundesministerium  
für Wirtschaft  
und Energie

aufgrund eines Beschlusses  
des Deutschen Bundestages

## Hinweis

Bei der projektausführenden Stelle kam es zu Verzögerungen bei der Besetzung der zugehörigen Stelle. Die Bearbeitung des Forschungsprojektes begann deshalb erst am 01.02.2018.

## Kurzfassung der Ergebnisse

Im Rahmen des Projektvorhabens bestand das Ziel in der Entwicklung einer Methode zur temporären Analyse intralogistischer Systeme speziell für klein- und mittelständische Unternehmen (KMU), die zum einen im Vergleich mit manuellen Zeitaufnahmen eine objektive, fundierte Analyse ermöglicht und zum anderen im Vergleich zu automatisierten Methoden, mit denen komplexe Analysen möglichst sind, einen reduzierten Installations-, Betriebs- und Investitionsaufwand bietet. Im Rahmen des Forschungsvorhabens wurde der Einsatz preiswerter, wiederverwendbarer Sender- (Beacon-Technologie) und Empfängertechnologie (wie bspw. Smartphones) in Verbindung mit der Entwicklung einer universellen Methode zur Erstellung und Auswertung von Bewegungsprofilen der zu untersuchenden Materialflusssysteme umgesetzt. Grundlage dafür sind die von einem bewegten Empfangsgerät, welches bspw. eine getracktes Flurförderzeug repräsentiert, empfangene Bluetooth Low Energy (BLE) Signale. Das BLE-Signal ist ein Funksignal, welches durch den sogenannten RSSI-Wert (received signal strength indicator) messbar ist und je nach Entfernung zum Empfänger variiert. Für die flexible Signaldatenaufnahme wurde eine Android-basierte Applikation entwickelt, welche es ermöglicht, neben der Signalaufnahme, Analyse-spezifische Einstellungen vorzunehmen und Prozessklassen sowie Layoutdaten zu hinterlegen. Die Datenbasis umfasst neben den RSSI-Signaldaten die Positionsdaten der bei der Systeminitialisierung hinterlegten Beacons und die durchgeführten Prozessklassen, welche die Grundlage zur retrospektiven Analyse der überwachten Prozesse bilden. Die dafür entwickelte Software-Architektur besteht aus fünf wesentlichen Elementen: Datenaufnahme, Datenaufbereitung, Prozessklassifizierung, Visualisierung der Bewegungsprofile und Auswertung. Die Datenaufbereitung ist notwendig, da das dem BLE zugrundeliegende Funksignal (RSSI) Schwankungen unterliegt. Durch gezielten Einsatz von Signalglättung und Interpolationsfiltern wird das Signal Szenario-abhängig aufbereitet. Mit Hilfe von Machine Learning-Algorithmen wird die automatisierte Zuordnung von definierten Prozessklassen zur Signaldatenbasis realisiert, um Auswertungen der betrachteten Prozesse zu ermöglichen. Zur Ableitung der Bewegungsprofile von Materialflussobjekten kommen spezifische Distanzkalkulationsfunktionen sowie Positionierungsalgorithmen zum Einsatz, die auf Basis der RSSI- und Positionsdaten die Zielstellung der Fixpunktreduzierung durch eine permanente Systemselbstkonfiguration umsetzen. Die Auswertung der generierten Prozessdaten erfolgt zum einen visuell über Bewegungsprofile und Prozessdiagramme sowie anhand logistischer Kennzahlen.

Das Ziel KMU zu befähigen, selbständig und mit wirtschaftlich geringem Aufwand Prozesse in flexiblen logistischen Systemen zu analysieren (bspw. zur Effizienz der Gabelstaplerrundfahrt), konnte realisiert werden. Somit kann eine objektive sowie fundierte Prozessanalyse als Basis für eine kontinuierliche Prozessverbesserung eingesetzt werden, um Abweichungen zu erkennen, Optimierungsstrategien abzuleiten und zu evaluieren. Die entwickelte Methodik unterstützt KMU zudem in der Generierung einer fundierten Datengrundlage auch ohne Installation umfangreicher Betriebsdatenerfassungs- oder -überwachungstechnologien, wie bspw. Track and Trace Systeme zur Auftragsverfolgung. Die Ziele des Forschungsprojektes wurden somit erreicht.

# 1. Durchgeführte Arbeiten im Berichtszeitraum

## (AP1) Klassifizierung von Prozessen in produktionslogistischen Systemen

### AP1.1 Kategorisierung intralogistischer Prozesse

Für eine gezielte Entwicklung der aufgezeigten Projektbestandteile ist die Ableitung des Anforderungsprofils grundlegend. Dazu zählt vor allem die Spezifizierung der logistischen bzw. produktionslogistischen Prozesse, die für eine Prozessanalyse der KMU Relevanz besitzen. Darauf basierend wurde ein Katalog mit Hilfe der Unternehmen des Projektbegleitenden Ausschusses (PA) abgeleitet, der potentielle Untersuchungsszenarien sowie Unternehmenscharakteristiken beinhaltet. Hierzu wurden im Rahmen des Kick-Off Meetings des Projektes zum einen ein Fragebogen ausgegeben und zum anderen Untersuchungsszenarien mit den Mitgliedern des PA diskutiert. In Tabelle 1 ist die Übersicht der fünf relevanten Untersuchungsszenarien zusammengefasst:

*Tabelle 1: projektrelevante Untersuchungsszenarien*

Durchlaufzeit Fertigungsauftrag
Materialflussanalyse Transportmittel (Durchlaufzeit, Transportspiele, Auslastung etc.)
Gabelstaplerrundfahrt (Kürzeste-Wege-Analyse)
Zonierung Lager

Nachfolgend erfolgten zudem noch Vorortbegehungen bei einzelnen Unternehmen, um die Erfassung von Materialflussobjekten sowie -prozessen und Untersuchungszielen umfassend zu ermöglichen. Im Ergebnis lag eine Kategorisierung möglicher Untersuchungsszenarien sowie der damit zusammenhängenden Materialflussprozesse, des benötigten Datenbedarfs sowie der aktuell verfügbaren Daten vor. Die Ergebnisse sind zudem in die AP3.1 und AP5.1 eingeflossen.

### AP1.2 Erfassung intralogistischer Untersuchungsziele

Mit Hilfe der in AP 1.1 erarbeiteten Anforderungen und der Übersicht über KMU-spezifische Untersuchungsszenarien konnte die Basis für die Ableitung von projektrelevanten Fragestellungen und dafür notwendige unternehmensbezogene Daten geschaffen werden. Es wurden in diesem Schritt ebenfalls relevante logistische Kenngrößen sowie die Datenbasis zur Bestimmung der prozessbezogenen Kennzahlen abgeleitet, kategorisiert und den jeweiligen Untersuchungszielen zugeordnet.

Im Ergebnis ergab sich eine Gesamtübersicht konkreter Untersuchungsziele, aus der für die Entwicklung der Analysemethode im Rahmen der Projektbearbeitung ein Referenz-Szenario abgeleitet wurde. Bei einem Großteil der im PA befragten Unternehmen lag der Fokus auf der Analyse der Performance ihrer Flurförderzeuge sowie deren Materialflussprozesse. Aus diesem Grund wurde die Prozesskette einer Gabelstaplerfahrt als Referenz zur Entwicklung der einzelnen Bausteine der Analyse-Methodik ausgewählt. Die Prozesskette eines Gabelstaplers beinhaltet zudem alle relevanten Bewegungsarten bzw. Prozesse, die für die Entwicklung einer Methode zur Prozessanalyse für die Logistik und Produktion relevant sind, wie bspw. geradlinige Bewegung auf ein Objekt zu oder davon weg. Die Untersuchung hinsichtlich eines sinnvollen Detaillierungsgrads und abschließende Definition der konkreten Prozessklassen (bzw. Bewegungsklassen) aus der Prozesskette erfolgen im Rahmen des vierten Arbeitspakets.

Im AP1 wurden zusammenfassend folgende Ergebnisse erzielt:

- Kategorisierung der Materialflussprozesse, Untersuchungsziele sowie Datenbedarf/ Datenverfügbarkeit
- Erkenntnisgewinn über Analysebedarfe

(AP2) Auswahl Sender- und Empfängertechnologie

Auf Basis der in AP1 definierten KMU-relevanten Untersuchungsziele sowie der Referenzprozesskette wurden Anforderungsprofile für die Sender- und Empfängertechnologie abgeleitet. Grundlegend bestand bereits zur Beantragung des Forschungsvorhabens die Zielstellung die Beacon-Technologie als Signalsender einzusetzen, da diese die Anforderungen einer kostengünstigen sowie flexiblen Technologie erfüllen. Ergänzend dazu wurde eine breite Recherche der am Markt verfügbaren Alternativen möglicher Signalsender auf Basis definierter Anforderungen und Leistungsparameter (siehe Tab. 2) untersucht (WLAN, Ultrabreitband etc.).

*Tabelle 2: Leistungsparameter Sendertechnologie*

Leistungsparameter	Anforderung
Protokoll	anpassbar: iBeacon, Eddystone
Eigene App/ Dashboard	Ja
Batterielebenszeit	mind. 1 Jahr, austauschbar
Sendeintervall	anpassbar: mind. 100 ms
Reichweite	mind. 70 m
Transmission Power	anpassbar
Empfindlichkeit	anpassbar
Preis	informativ, < 40,00 € Beacon
Add-on	Nein

Wesentliche Leistungsparameter für den Einsatz im Rahmen des Analyse-Tools sind die Signalreichweite, Batterielebensdauer, verwendete Signalprotokolle sowie Kosten der Beacons. Zudem müssen die Beacons möglichst einfach konfigurierbar sein, um zum einen eine Anpassung im Rahmen der Projektbearbeitung und zum anderen an unterschiedliche Untersuchungsszenarien flexibel ermöglichen zu können. Die meisten Beacon-Hersteller stellen eine eigens für die Beacon-Konfigurierung entwickelte App bzw. ein Web-basiertes Interface mit individuellen Zugangsdaten zur Verfügung, über welche eine Konfiguration erfolgen kann. Wesentlich für einen flexiblen Einsatz sind die Anpassbarkeit des Sendeintervalls, des Sendeprotokolls mit den enthaltenen Informationen und die Anpassbarkeit des tx-Powerwerts (Referenzsignalstärke bei 1m) durch den Anwender. Während der Technologie-Auswahl war die Nutzung eines konkreten Sendeprotokolls noch offen, weshalb es notwendig war, Hersteller auszuwählen, deren Beacons es ermöglichten, verschiedene Protokolle einzustellen sowie die je Protokoll mitgegebenen Informationen (z.B. Beacon-ID) anzupassen.

Nach umfassender Recherche konnte eine Auswahl von fünf Beacon-Herstellern erfolgen. Zu Testzwecken wurden entweder von den Herstellern angebotene Starter-Kits oder eine geringe Anzahl an Test-Beacons beschafft und ausgiebige Tests auf Basis der o.g. Anforderungen durchgeführt. Für die weitere Verwendung konnten zwei Hersteller identifiziert werden, die zum einen preislich attraktive und umfangreich konfigurierbare Beacons anbieten sowie zum anderen eine einfache Konfiguration via Web-Interface und/oder App ermöglichen.

Für die Auswahl der Empfängertechnologie bestand grundlegend die Anforderung an eine Bluetooth-Schnittstelle, die das BLE-Signal empfangen kann. BLE basiert auf dem Bluetooth-Standard 4.0, somit waren alle Empfangsgeräte mit einer Schnittstelle ab Bluetooth 4.0

kompatibel. Die Empfängertechnologie sollte zudem Smart Mobile Devices (SMD) umfassen, um die Anforderung einer flexiblen Einsetzbarkeit sowie eines geringen monetären Aufwands zu fokussieren. Im Rahmen der Marktrecherche von infrage kommender Technologie erfolgte eine Eingrenzung auf Smartphone mit Android-basiertem Betriebssystem ab Android 6.0, Raspberry Pis sowie Tablets. In der ersten Testphase wurden Smartphones unterschiedlichster Anbieter sowie ein Raspberry Pi hinsichtlich Empfang von BLE-Signalen mit kostenlos verfügbaren Beacon-Scanner-Anwendungen getestet. Für das Raspberry Pi wurde die entsprechende Software selbst entwickelt und getestet.

Während der fortschreitenden Projektbearbeitung kam es immer wieder zu Auffälligkeiten während der Datenaufnahme bzw. bei der Datenaufbereitung, sodass sowohl das Raspberry Pi als auch ein Großteil der getesteten Smartphones als ungeeignet eingestuft werden mussten. Der Großteil der Android-basierten Smartphones haben aufgrund des Betriebssystems den Schwachpunkt, dass das Betriebssystem auf im Hintergrund laufende Anwendungen zugreift und diese autonom beendet, um den Energiespareinstellungen gerecht zu werden. Dies konnte nicht mit entsprechender Anpassung der Einstellungen behoben werden, auch nicht durch die Deaktivierung von Systemanwendungen. Zudem konnte festgestellt werden, dass jedes Smartphone unterschiedliche Schnittstellentechnologien (z.B. Bluetooth-Connectoren) aufweist sowie die interne Signalverarbeitung Technologie-abhängig ist. Somit erwiesen sich lediglich die Smartphone-Generation eines Herstellers als geeignet, um dauerhafte (länger als 30 min) Datenaufnahmen zu gewährleisten. Die Einbindung des Raspberry Pis zur Datengenerierung hat sich im Fortschreiten der Projektbearbeitung als sehr aufwändig und nicht praktikabel erwiesen. Für den Einsatz des Einplatinen-Computers mussten die Software-Elemente selbst entwickelt und getestet werden. Zudem waren ein separater Bildschirm sowie eine separate Energieversorgung notwendig. Im Projektverlauf wurde sich deshalb auf die Weiterverwendung der Smartphones zur Datengenerierung konzentriert.

Im Zusammenhang mit der Auswahl der eingesetzten Technologie zur Datenaufnahme stand ebenfalls die Fragestellung, wie das Datenmanagement erfolgen soll. Vor allem im Hinblick auf die folgenden Entwicklungsschritte der Prozesserkennung sowie der Positionsbestimmung und Ableitung sowie Visualisierung der Bewegungsprofile musste ein geeignetes Verfahren zum Datenmanagement gefunden werden. Die Konzeptentwicklung sah die Umsetzung in Form einer MongoDB NoSQL-Datenbank vor. Die Datenbank konnte erfolgreich im Python-basierten Software-Framework eingebunden und getestet werden.

Im AP2 wurden zusammenfassend folgende Ergebnisse erzielt:

- Festlegung und Anschaffung der technisch-technologischen Anforderungen für das Analyse- und Messsystem (Beacons, Smartphones, Raspberry Pi)
- Umsetzung des Datenmanagementkonzeptes in Form einer SQLite / NoSQL Datenbank

### (AP3) Erstellung experimentelles Umfeld und Experimentdurchführung

Aus den Gesprächen mit den Unternehmen des PAs und den Anforderungen aus der Industrie wurden bereits in AP1 Untersuchungsziele und -szenarien definiert, auf deren Basis die Referenz-Prozesskette für die weiteren projektbezogenen Entwicklungen abgeleitet wurde. Im Folgenden wird aufgezeigt, wie sich im Rahmen des AP3 durch definierte Testumgebungen und umfangreichen Versuchen zu differenzierten Annäherungsbestimmung und Triangulation der sich relativ zueinander bewegenden Materialflussobjekte der Positionsbestimmung sowie der



Empfohlene Daten für die Untersuchung			
A) Prozesse des Materialflusses	Planungsprozesse		D) Daten vorhandener Fördermittel und Lageranlagen
	Beschaffungsprozesse	*	Kapazitäten (Fassungsvermögen, Lagerplätze)
	Herstellungsprozesse	*	Taktzeiten, Spielzeiten, Bedienzeiten, Reaktionszeiten, Greifzeiten
	Lieferungsprozesse	*	Durchsätze, Flächen- und Raumbedarf
	Rücklieferungsprozesse	*	Steuerungshardware/-software
B) Informationsverarbeitungssysteme	Lagerverwaltungssysteme		E) Produktionstechnische Daten
	PPS-/ERP-/APS-Systeme		Produkte (Stückzahlen, Losgrößen)
	Personalverwaltungssysteme		Produktion (Fertigungstechnik, -organisation)
	Finanzbuchhaltungssysteme		Aufträge (DLZ, Bestände, Terminplanung)
C) Daten des Materials	Abmessungen, Volumina, Gewichte	*	F) Daten des externen Warentransports
	Mengen (Durchsatz, Ankunftsrate, Verteilungsfunktion)	*	Lieferanten-/Kundenbeziehungen (Mengen, Entfernungen)
	Kennzeichnungen (Nummerierung, Codierung)		G) Betriebswirtschaftliche Daten
	Materialmerkmale (stoß-, druck-, schmutzempfindlich, begrenzt lagerfähig, explosionsgefährdet)		Investitionen, Kapitalbindungskosten
	Verwaltung der Bestände, der Zu- und Abgänge	*	Betriebskosten (Energie, Wartung, Personal)
	Ladungsträger, Möglichkeiten der Stapelbildung	*	Bilanz (Gewinn/Verlust)
			H) Sonstige Parameter
			Gebäude, Grundstücke, Unternehmensstrategie, Layouts
			Allgemeine Auflagen (Gesetze, Tarifverträge, Verordnungen, Richtlinien, Normen)

zurück zur Checkliste Quelle: Eigene Darstellung, i. A. a. Arnold/Furmans 2009: 235 f.

Abbildung 3: Empfehlung Szenario-basierter Datenbedarf im Versuchstemplate

Während der einzelnen Entwicklungsphasen der Analyse wurde eine Vielzahl von Testumgebungen generiert, die jeweils zur Beantwortung bestimmter Fragestellungen bzw. zur Generierung bestimmter Datensets konfiguriert wurden. Ergänzend dazu wurden bei den Mitgliedern des PA ebenfalls Testumgebungen eingerichtet und Versuche durchgeführt, wofür Anpassungen im Prozessumfeld notwendig waren (Details siehe AP3.5).

### AP3.2 Systemvorkonfiguration Sender und Empfänger

Die Systemvorkonfiguration bezieht sich vor allem auf die Sender- und Empfängertechnologie, wobei der Schwerpunkt der Entwicklungsarbeit in der Entwicklung einer anforderungsgerechten Datenaufnahme und -aufbereitung lag.

Zur Gewährleistung einer geeigneten Datenaufnahme der RSSI-Signalwerte sowie deren Weiterverarbeitung wurden im ersten Schritt Anforderungen definiert, die eine mobile Applikation erfüllen musste (siehe Tab. 3).

Tabelle 3: Anforderungen Signaldatenerfassung

<b>Betriebssystem</b>	Android
<b>Erfassungsfrequenz</b>	anpassbar, min. 100 ms
<b>Signalglättung</b>	Moving Average AutoRegressive Moving Average (Moving Average)
<b>Senderanzahl</b>	mind. 50
<b>Anzeigedaten (sortierbar)</b>	in Echtzeit: RSSI, evtl. Distanz je Beacon (ID: MAC-Adresse)
<b>Datenspeicherung</b>	Auslesen als .csv-Datei (inkl. Zeitstempel, RSSI je Beacons)

Darauf aufbauend erfolgte eine umfassende Marktrecherche vorhandener Beacon-Scanner-Apps für Android-Betriebssysteme sowie ein Test der Apps auf Basis der definierten Anforderungen. Die getesteten Anwendungen waren alle kostenlos verfügbar. Es konnte keine App identifiziert werden, die den Anforderungen gerecht wurde. Vor allem die gezielte Speicherung aufgenommener RSSI-Signalwerte war durch keine der Anwendungen möglich. Für einige Anwendungen gab es die Möglichkeit, im Entwicklermodus Anpassungen vorzunehmen, jedoch waren diese nicht so umfassend umsetzbar, wie es für die Anwendbarkeit im Rahmen der Projektbearbeitung notwendig gewesen wäre. Zudem war die Nachvollziehbarkeit verwendeter Algorithmen, z.B. Datenfilter oder Distanzkalkulation, in den bereits bestehenden Applikationen meist nicht gegeben, was für eine spätere Anwendung bzw. Fehlerdiagnose hinderlich gewesen wäre. Im Ergebnis musste festgestellt werden, dass eine eigene projektspezifische Entwicklungsumgebung einer Android-basierten Anwendung geschaffen werden musste. Die

App-Entwicklung erfolgte mit der Programmiersprache Java sowie basierend auf dem iBeacon-Standard für die BLE-Signalübertragung. Ausgehend von den definierten Anforderungen wurden die App-Funktionalitäten während der Projektbearbeitung sukzessive erweitert, an ergänzende Anforderungen während der Bearbeitung der AP 3, 4 und 5 sowie nach eingehenden Test der Usability durch Mitglieder des PA und durch Projektbearbeiter angepasst. So lag zu Beginn der Fokus auf der korrekten Aufnahme und Speicherung der RSSI-Werte sowie der Beacon-Informationen. Des Weiteren lag ein wesentlicher Schritt in der Auswahl, Implementierung und in den Tests der Signalglättungs- sowie Distanzkalkulationsalgorithmen sowie deren Einbindung in die Datenspeicherung. Die Integration von Szenario-basierten Informationsdaten, wie bspw. Layout-Bilddateien, Layoutmaßen, Aufnahmeorte etc., stellten eine weitere Anpassungsschleife dar (siehe Abb. 5). Zur Realisierung der Positionsbestimmung sowie der Visualisierung von Bewegungsprofilen wurde die Eingabe von Primärpositionen je Beacon sowie für das Empfangsgerät selbst implementiert. Zudem kann der Bediener festlegen, ob ein Beacon beweglich oder fix ist (siehe Abb. 4). Eine weitere wesentliche Erweiterung erfolgte für die Aufnahme von Trainingsdaten zum Anlernen der Machine Learning-Algorithmen durch die Implementierung von Prozessklassen. Für definierte Standardklassen wurden Schaltflächen zur intuitiven Eingabe der Prozessklassen während einer Datenaufnahme vorgesehen und zudem besteht die Möglichkeit, weitere Prozessklassen je nach Untersuchungszenario zu hinterlegen (siehe Abb. 4). Die gesamte Übersicht über die einzelnen Schritte der App-Entwicklung ist im Anhang zu finden. Tabelle 4 beinhaltet eine Übersicht der App-Spezifikationen.

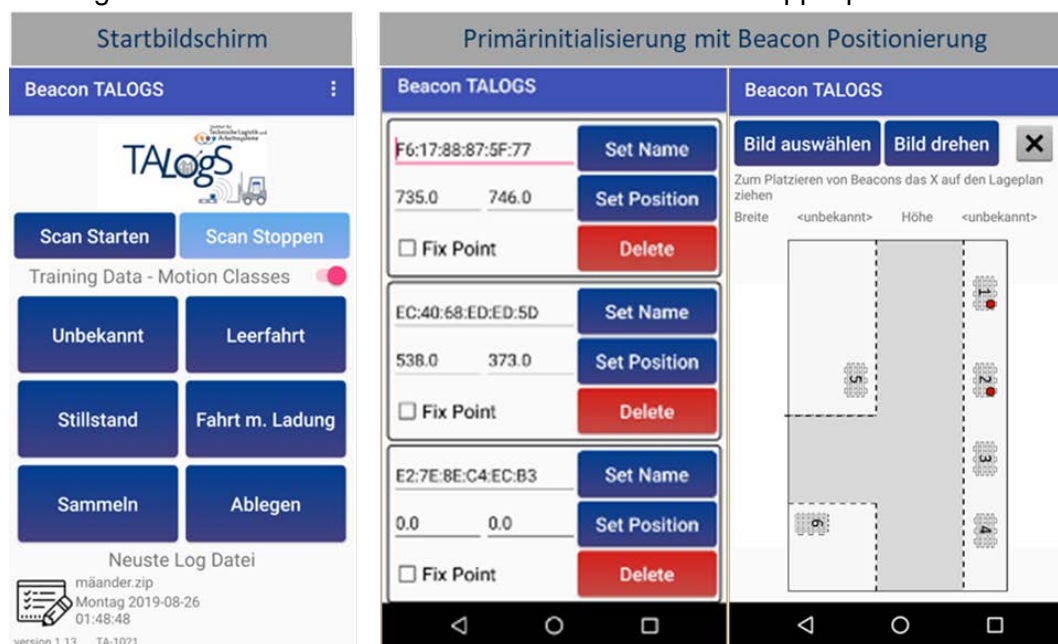


Abbildung 4: Funktionalitäten TALogs-App I



Einstellmöglichkeiten		Echtzeitdaten-Anzeige
<b>Beacon TALOGS</b>	<b>Beacon TALOGS</b>	<b>Beacon TALOGS</b>
<b>Geräte Name</b> eindeutiger name für dieses gerät	<b>Bewegungsklassen</b> Bewegungsklassen für Trainingsdatenaufnahme	<b>Bluetooth Addr. RSSI Age (ms)</b> C2:D2:6C:30:C9:B0 -76 1036 C4:A2:6D:7C:38:0D -74 5 C5:CB:B5:35:EC:21 -70 7 CE:10:92:5D:27:79 -74 8 DC:19:D3:2A:F5:AC -75 10 DC:F1:41:90:88:61 -64 12 DD:C4:55:11:2A:03 -69 <b>2064</b> E2:20:87:77:BF:F9 -74 15 E7:4A:5A:8E:64:1A -72 1051 E9:8B:D0:8F:68:5C -70 19 EC:40:68:ED:ED:5D -74 20 F6:17:88:87:5F:77 -73 21 F7:50:F1:DC:80:9D -64 1054 FB:D7:30:BA:17:3B -75 22 FC:0F:2F:0A:6E:FF -70 1056
<b>Aufnahmeort</b> eindeutiger name für den aufnahmeort	<b>Debug Info</b> <input type="checkbox"/> zeige debug info während der aufnahme	
<b>Sitzungsname</b> eindeutiger name für die sitzung	<b>All Beacon Einstellungen</b>	
<b>Csv Split Periode</b> sekunden bis eine neue csv datei angefangen wird	<b>Scan Periode</b> millisekunden zwischen scans	
<b>Filter Beacons nach Id</b> <input type="checkbox"/> nur beacons aufnehmen, die in der initialisierung enthalten sind	<b>RSSI Filter</b> <a href="https://altbeacon.github.io/android-beacon-library/distance_vs_time.html">https://altbeacon.github.io/android-beacon-library/distance_vs_time.html</a>	
<b>Bewegungsklassen</b> Bewegungsklassen für Trainingsdatenaufnahme	<b>Running Average Fenstergröße</b> fenstergröße von mittelwert in millisekunden	
	<b>ARMA Faktor</b> <a href="https://altbeacon.github.io/android-beacon-library/distance_vs_time.html">https://altbeacon.github.io/android-beacon-library/distance_vs_time.html</a>	

Abbildung 5: Funktionalitäten TAlOGS-App II

Tabelle 4: Implementierte App-Spezifikationen

<b>Betriebssystem</b>	Android
<b>Erfassungsfrequenz</b>	anpassbar, min. 100 ms
<b>Signalglättung</b>	Moving Average AutoRegressive Moving Average (ARMA)
<b>Senderanzahl</b>	mind. 50
<b>Erfassungsdaten</b>	Zeitstempel RSSI registrierte Beacons (mit ID (MAC-Adresse), fix oder variabel) Primärkoordinaten Sender Primärkoordinaten Empfänger Prozessklassen
<b>szenario-basierte Einstellungen</b>	Layoutdaten, inkl. Maßstab Filtereinstellung Beaconzuordnung Szenario-Informationen (Projektname, Aufnahmeort etc.)
<b>Anzeigedaten</b>	Rohdaten (Echtzeit-RSSI-Werte) gespeicherte Log-Files
<b>Datenspeicherung</b>	<u>zip-Datei bestehend aus:</u> Rohdaten-File .csv (Zeitstempel, Beacon mit RSSI, Distanzwert, geglätteten RSSI) Log-File .json (Layoutmaße, Primärkoordinaten, Empfangsgerät, Aufnahmedaten, Filtereinstellung)

Nach jeder Entwicklungsschleife wurden die App-Funktionalitäten sowie die Leistungsfähigkeit umfassend getestet. Dies erfolgte zum einen durch Datenaufnahmen in entsprechenden Testszenarien und zum anderen durch Prüfung der ausgegebenen Rohdaten-Dateien.

Für die Aufbereitung der via App aufgenommenen Signaldatenbasis wurde ein weiteres Modul des Software-Frameworks für die Analysemethode entwickelt. Das Python-basierte Converter-Skript liest die generierte Rohdaten-Datei als Daten-Framework (Python-Package pandas) ein und wandelt die Rohdatenformate in lesbare Formate um, wie bspw. Timestamp zu Datetime. Die empfangenen Datenpakete der RSSI-Werte werden im Skript auf eine 100stel Sekunde gemittelt, mit Hilfe der nachfolgend aufgezeigten Methoden aufbereitet und alle weiteren Datenformate (Distanzdaten (AP3.4) etc.) werden davon abgeleitet. Sind Prozessklassen hinterlegt worden,

werden diese ebenfalls aufbereitet und in einer Datenspalte mitgeschrieben. Die implementierten Methoden zur Signalaufbereitung umfassen den gleitenden Durchschnitt, der über Anpassung im Skript variiert werden kann, sowie verschiedene Interpolationsmethoden, wie bspw. Polynomfilter, Linearfilter oder kubische Filtermethode. Für die Datenglättung wurde zudem der Ansatz der outlier detection im Skript hinterlegt. Auf Basis des Medianwerts sowie eines Vielfachen der Standardabweichung der RSSI-Datensets je Beacons werden die Ober- und Untergrenzen für die Ausreißerererkennung und -beseitigung ( $Grenzwert = \bar{x} \mp x * \sigma$ ) bestimmt. Die Methoden können wie der gleitende Durchschnitt im Skript ausgewählt und angepasst werden. Ergebnis des Skriptes ist die Ausgabe der aufbereiteten Daten in einer Excel-Datei, die für weitere Analyse nutzbar ist. Im Rahmen der Entwicklung des Converter-Skripts wurde die Analyse der aufgenommenen Daten hinsichtlich Eignung der implementierten Datenglättungs- sowie Interpolationsmethoden sowie deren Parametrierung fokussiert, um der Zielstellung einer Szenario-basierten Signalaufbereitung näher zu kommen. Eine allgemeingültige Ableitung von Parametereinstellungen zur bestmöglichen Signalaufbereitung je untersuchtem Szenario war nicht möglich. Es konnte lediglich eine Einschränkung geeigneter Verfahren sowie deren Parametersets erfolgen (siehe Tab. 5).

*Tabelle 5: Parametersets zur Datenerfassung und -konvertierung*

Parameter	Hinterlegung	Variation
Rolling Mean	App	50, 100, 1000 [ms]
Scan Periode	App	10, 100, 500, 1000 [ms]
Rolling Mean	Converter-Skript	1, 3 [sek]
Interpolation	Converter-Skript	cubic, linear, polynomial, spline [Order 3, 5]
Ausreißer	Converter-Skript	Faktor $x$ von $\sigma$
Distanzkorrekturfaktor	Converter-Skript	$a, b$ SMD-spezifisch

Abschließend wurde das Modul zur Datenaufbereitung an die bereits bestehende Datenbank angebunden, sodass die Möglichkeit besteht die aufbereiteten Daten sowohl direkt in die Datenbank zu übergeben als auch in das Excel-Format zu konvertieren.

### AP3.3 Einfache Annäherungsbestimmung

Aufgrund der zugrundeliegenden Ausbreitungsmodelle für Funk- bzw. elektromagnetische Wellen unterliegt das BLE-Signal oder die RSSI-Daten Schwankungen, welche durch in der Umgebung auftretenden Dämpfungseffekten hervorgerufen werden. In der Projektbearbeitung wurden für die Signalausbreitung die Annahmen des log-distance path loss Modell (Evennou/Marx (2006)) übernommen, welches einen logarithmischen Zusammenhang zwischen Signalstärke und Entfernung vorgibt. So nimmt das empfangene Signalwerte (RSSI) mit zunehmender Entfernung zwischen Sender und Empfänger ab:

$$RSSI(d) = RSSI(d_0) - 10 * \alpha * \log_{10}\left(\frac{d}{d_0}\right)$$

Dabei stellt  $RSSI(d)$  die empfangene Signalstärke bei der gesuchten Entfernung  $d$  dar.  $RSSI(d_0)$ , eine Referenzmessung bei der Entfernung von einem Meter, ist die empfangene Signalstärke bei der bekannten Entfernung  $d_0$  zum Sender und  $\alpha$  ist ein Koeffizient, der die Ausbreitung der Wellen in der untersuchten Umgebung modelliert und je nach Art und Einflussfaktoren in der Umgebung variiert. Einflussfaktoren auf die Signalstärke haben die Dämpfungseffekte Reflexion, Beugung, Streuung und Absorption. Vor allem im Produktions- und Logistikumfeld sind in einem hohem Maße Störquellen vorhanden, die zu Signaldämpfungen führen.

Auf Basis dieser Annahmen lag das Ziel dieses AP in der Bewertung und Klassifizierung vorhandener Störgrößen sowie die ggf. notwendige Ableitung von Korrekturkoeffizienten zur Reduzierung der Dämpfungseffekte auf die Signaldatenbasis, um eine möglichst korrekte Transformation der Signalstärke in Abhängigkeit von Hindernissen in Entfernungsdaten zu gewährleisten. Entfernungsdaten bilden die Grundlage für die nachfolgende Positionsbestimmung zur Abbildung der Bewegungsprofile getrackter Materialflussobjekte. Im ersten Schritt wurde ein experimenteller Ansatz entwickelt, der die Planung und Umsetzung zielgerichteter Versuche inkl. Versuchsdauer, -aufbau, -wiederholungen etc. zur Bestimmung der umgebungsabhängigen Dämpfungskoeffizienten  $\alpha$  vorsah. Es wurden umfangreiche Experimente in verschiedenen produktionsnahen Testumgebungen durchgeführt, wie bspw. einer Produktionshalle sowie in einer Holzmanufaktur, zum einen mit ruhenden und zum anderen mit dynamischen Sendern und Empfängern. Untersuchungsfokus waren typische Störquellen, die in Logistik- sowie Produktionshallen vorhanden sein können – hier sind z.B. zu nennen: Maschinen, Wände sowie Türen aus verschiedenen Materialien, Menschen und andere Signalarten (WLAN, GPS, RFID etc.). Insbesondere für Stahltüren (Beispiel siehe in Abb. 6) bzw. Beton mit Stahlbewehrung und Aluminium konnte eine hohe Signaldämpfung nachgewiesen werden (siehe Abb. 7).

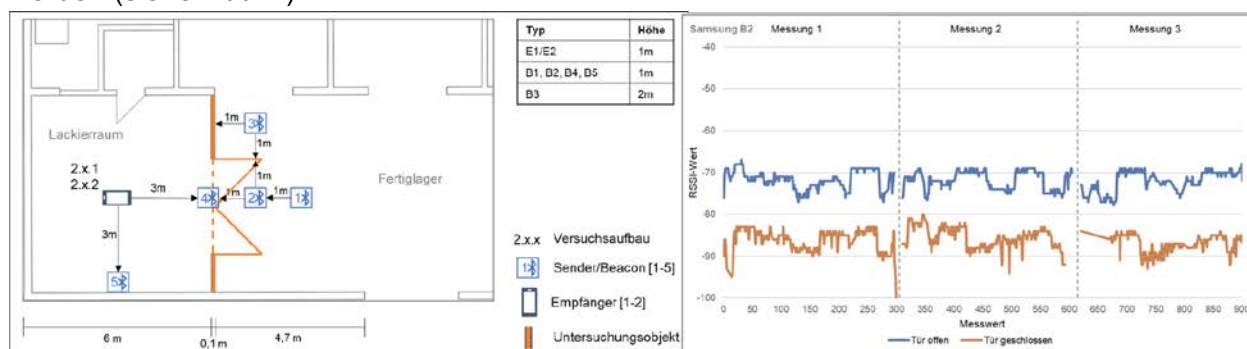


Abbildung 6: Versuchsaufbau und Signalstärke-Messdaten für Störquelle Stahltür

Dämpfeigenschaften	Material	Ø Dämpfung
schwache (0 bis -7,33)	Ziegel	-0,62
	Holz	-2,00
	Glas	-3,12
mittlere (-7,34 bis -14,66)	Porenbeton	-8,24
	Kalksandstein	-9,65
starke (-14,67 bis -22)	Stahl	-15,43
	Mensch	-17,23
	Porenbeton mit Stahlträger	-17,34
	Aluminium	-20,64

Abbildung 7: Einordnung Dämpfungseigenschaften ausgewählter Störquellen

Trotz der Erkenntnisse über die unterschiedlichen Dämpfungseigenschaften der Störquellen war es nicht möglich, ein allgemeingültiges Korrekturmodell der Signaldämpfung auf Basis des o.g. Ausbreitungsmodell abzuleiten, da die generierten Dämpfungskoeffizienten je Testumgebung sehr unterschiedlich ausgefallen sind. Die in der Literatur (Vgl. z.B. Vaughan/ Andersen (2003)) häufig angenommenen Standardwerte bzw. Durchschnittskoeffizienten für die Ableitung der Entfernung auf Basis des o.g. Ausbreitungsmodells konnten im Rahmen der Projektbearbeitung

keine zufriedenstellenden Ergebnisse für die Distanzbestimmung hervorbringen. Die rechnerisch bestimmten Distanzwerten wichen zu stark von den physisch gemessenen Distanzen ab.

Bei der Auswertung der durchgeführten Experimente ist zudem auffällig gewesen, dass die verwendeten Smartphones unterschiedliche Ergebnisse hinsichtlich der empfangenen Signalstärkedaten ausgaben (siehe Abb. 8). Dies bestätigte die Vermutung, dass die Signalverarbeitung abhängig von den Geräteschnittstellen sowie der Gerätesoftware ist. Die auffällige Smartphone-abhängige Signalverarbeitung wurde in weiteren Experimenten bestätigt.

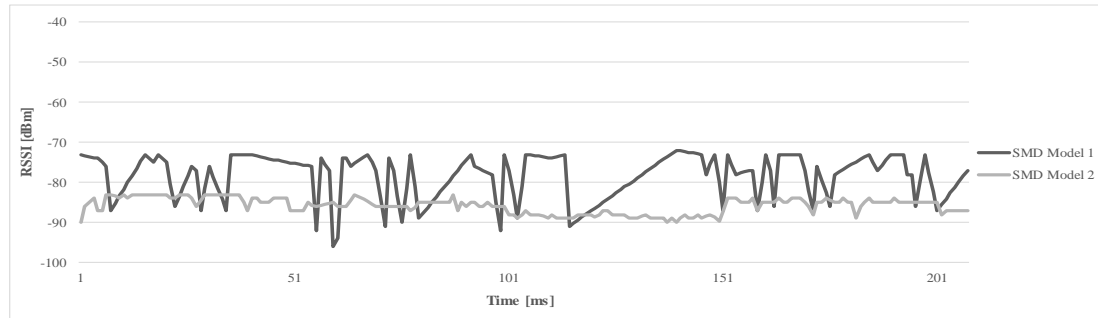


Abbildung 8: SMD-abhängige Signalstärkewerte

Diese Erkenntnis erschwerte jedoch zusätzlich die Ableitung einer allgemeingültigen Standardfunktion zur Distanzbestimmung. Darauf aufbauend erfolgte eine sehr umfassende Datenaufnahme zur Ableitung Smartphone-abhängiger Distanzkalkulationsfunktionen bzw. -koeffizienten. Anders als in den Vorversuchen wurde der Fokus nicht auf die Einflüsse einzelner Störgrößen und Dämpfungsfaktoren auf die Signalstärkedaten gelegt, sondern der Ansatz der Altbeacon Library (Android Beacon Library (20219)) zur Distanzkalkulation verfolgt. Das Android Beacon Library Modell basiert auf einem Verlustfaktor, welcher in nicht logarithmierter Form vorliegt. Der Ansatz dieses Modells liegt darin, dass die RSSI-Werte und Leistungsverluste sender- und empfängerspezifisch sind. Zur individuellen Distanzkorrektur wurde im Rahmen der Untersuchungen für jedes während der Entwicklung verwendete SMD eine Annäherungsbestimmung (Messung Entfernung  $IST_{\text{physisch}}$  zu  $IST_{\text{Signal}}$ ) durch Transformation der RSSI-Werte in Distanzwerte vorgenommen. Die Bestimmung der Koeffizienten erfolgte in einer idealisierten Umgebung, um andere Einflussgrößen, wie andere Signalarten, dämpfende Materialien, o.ä., zu minimieren und somit ein möglichst genaues und gerätespezifisches Distanzkorrekturmodell zu erhalten. Die Annäherungsbestimmung erfolgte auf einer ebenen Freifläche ohne umschließende Bebauung sowie abseits von städtischen und industriellen Einflüssen. Es wurden je SMD und je Distanz (0,25 m – 40,00 m) je drei Messungen (je 60 sek) aufgenommen. In Abbildung 9 ist beispielhaft ein Kurvenverlauf für die Annäherungsbestimmung eines SMD dargestellt. Es ist deutlich erkennbar, dass mit zunehmender Distanz zwischen SMD und Beacon die RSSI-Werte stärker schwanken.

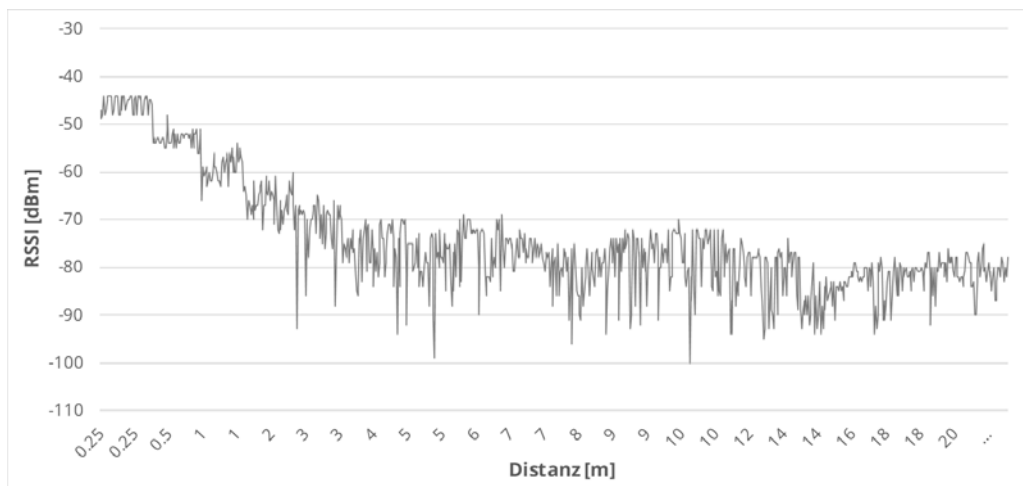


Abbildung 9: Kurvenverlauf Annäherungsbestimmung Beispiel-SMD

Mit Hilfe einer Curve-Fitting-Funktion wurden die SMD-individuellen Korrekturkoeffizienten abgeleitet (siehe Abb. 10). Diese sind im entsprechenden Software-Modul der Datenaufbereitung sowie in der Datenbank hinterlegt und können je verwendeten SMD anhand der gerätespezifischen ID ausgewählt werden.

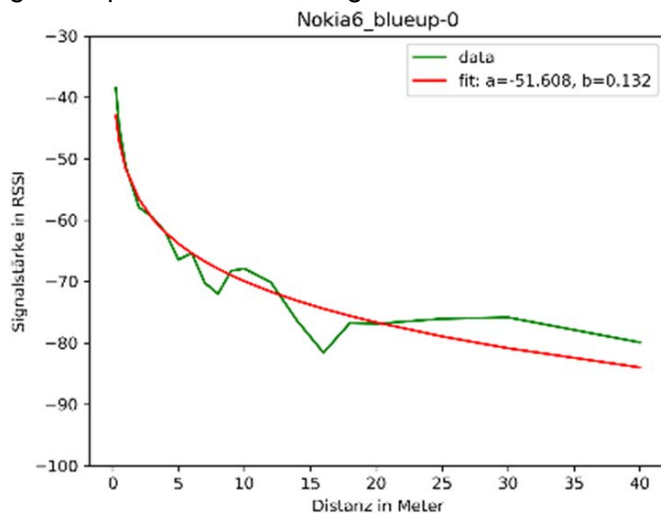


Abbildung 10: Ergebnisdarstellung Curve-Fitting-Funktion

Die fehlerhafte Distanzkalkulation konnte für die bisherigen Versuchsaufbauten der Realität sowie Empfangsgerät-spezifisch angenähert werden.

### AP3.4 Komplexe Annäherungsbestimmung

Die übergeordnete Zielstellung dieses AP war die Gewährleistung einer retrospektiven und hinreichend genauen Bestimmung von Positionsdaten als Grundlage für die Ableitung von Bewegungsprofilen auf Basis der aufgenommenen Signalstärkedaten sowie der daraus bestimmten Distanzdaten und der bei der Initialisierung hinterlegten Koordinaten je Beacon und Empfangsgerät. Als weitere ergänzende Meta-Informationen sollten lediglich die Layoutdaten der jeweiligen Untersuchungsszenarien herangezogen werden. Ein weiterer Entwicklungsfokus war die Realisierung einer Positionsbestimmung auch ohne vorhandene stationäre Beacons, das heißt ohne vorhandene Fixpunkte. Eine weitere Herausforderung bestand darin, für die Visualisierung von Bewegungsprofilen zusätzlich die Informationen zu hinterlegen, ob ein Transport stattgefunden hat oder nicht.

Auch für die Positionsbestimmung wurde ein eigenes Software-Modul in das System-Framework integriert, das ebenfalls auf Python basiert und unterschiedliche Methoden des Position Processings integriert. Die Kernfunktionalität des Positionierungsmoduls ist die Berechnung der Positionsveränderungen von Sendern und einem aber auch mehreren Empfängern im zeitlichen Verlauf basierend auf bekannten sowie übertragenen Initialisierungspositionen. Zur Erreichung der Positionsbestimmung ohne stationäre Beacons wurde der Ansatz verfolgt, dass die Position des Empfangsgerätes dazu genutzt wird, die Senderpositionen neu zu kalibrieren. Auf Basis der euklidischen Distanz zwischen Empfangsgerät und den drei nächsten Sendern wird mit Hilfe der Trilateration die Berechnung der Neupositionspunkte ( $P_{D_i}'(t)$ ) durchgeführt. Ein Prüfalgorithmus stellt sicher, dass die Neupositionen nicht innerhalb eines nichtplausiblen Bereichs im Layout liegen, z.B. einem Maschinenstandort oder einem Lagerregal (siehe Abb. 11).

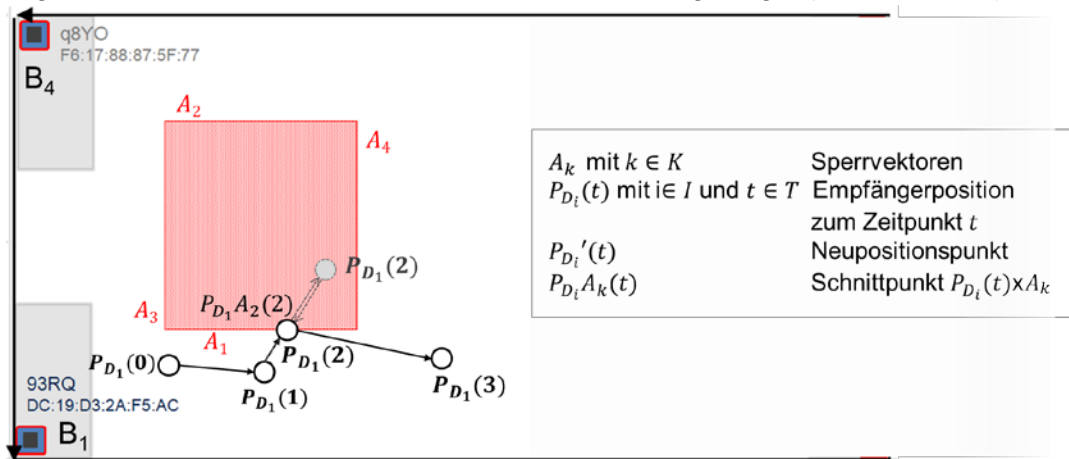


Abbildung 11: Schema Positionskorrektur

Es wird geprüft, ob die Gerade (Richtungsvektor) zwischen dem vorherigen Punkt  $P_{D_i}(t - 1)$  und dem angenommenen Neupunkt  $P_{D_i}'(t)$  eine der Kanten (repräsentiert durch die Sperrvektoren  $A_k$ ) des nichtplausiblen Bereichs schneidet. Wird dies festgestellt, erfolgt die Korrektur des Neupositionspunktes auf den Schnittpunkt ( $P_{D_i}A_k(t)$ ) zwischen Sperrvektor und Gerade. Somit wird erreicht, dass die Richtung der Bewegung erhalten bleibt, jedoch an den bestehenden Grenzen endet. Der komplette sowie detailliert dargestellte Funktionsumfang des Algorithmus zur Positionsbestimmung ist als Anlage zu diesem Dokument gegeben.

Die Entwicklung des dargestellten Algorithmus erfolgte sukzessive über den gesamten Zeitverlauf der Projektbearbeitung. In den einzelnen Entwicklungsphasen wurden die Algorithmusergebnisse durch umfangreichen Versuchsdaten, welche sowohl konkret zum Test des Algorithmus durchgeführt wurden als auch aus Realversuchen stammen, geprüft. Durch die Anbindung an die entwickelte GUI (AP5.5) konnte ebenfalls eine visuelle Prüfung der korrekt bestimmten Positionsdaten anhand der Bewegungsprofile erfolgen.

### AP3.5 Szenarienbasierte Experimentdurchführung

Das angestrebte Experimentdesign in diesem AP wurde auf Basis des in AP3.1 entwickelten Versuchstemplate umgesetzt. Das Template enthält für eine Versuchsplanung und -durchführung alle relevanten Informationen. Die Versuchsdurchführung wurde zum einen grafisch in Layoutdokumenten, handschriftlich zum anderen durch eine aussagekräftige Fotodokumentation festgehalten. Im Rahmen des AP erfolgten zudem prototypische Implementierungen bei zwei Unternehmen aus dem PA, um das Versuchstemplate sowie die Funktionsfähigkeit des Gesamtsystems zu testen. Eine konkrete Zielstellung war in einem der beiden Feldversuche die Generierung von Datensätzen zur Analyse der Materialflussprozesse mehrerer Gabelstapler

eines Kommissionierbereichs. Layoutdaten lagen im Rahmen der Projektbearbeitung nicht vor und mussten händisch erstellt werden. Zudem war es notwendig, den zu analysierenden Bereich räumlich einzugrenzen, da die durch die Gabelstapler bediente Fläche um ein Vielfaches größer war als der eigentliche Kommissionierbereich. Unternehmensseitige Untersuchungsziele waren zum einen die Anteile an Leerfahrten, Rangier- und Transportzeiten (basierend auf den Prozessklassen – siehe AP4) sowie zum anderen die Analyse der Transportspiele (siehe Abb. 11).

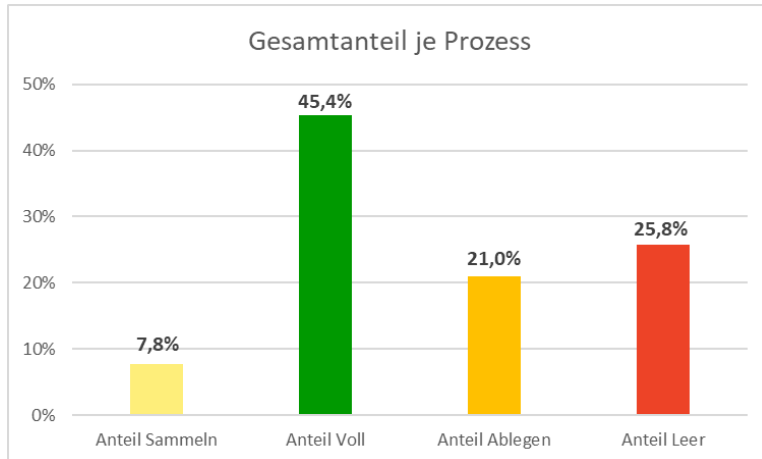


Abbildung 11 Anteil Prozessklassen bei Feldversuch

Projektseitig lag der Fokus auf der Erprobung der Benutzerfreundlichkeit und Leistungsfähigkeit der entwickelten Smartphone-Applikationen zur Datenaufnahme (AP3.2) sowie die Visualisierung der Bewegungsprofile der Gabelstapler mit Hilfe des entwickelten Demonstrators (AP5). Ähnliche Anpassungen sowie Erarbeitung einer Prinzipskizze war für eine Testumgebung bei einem weiteren Unternehmen aus dem PA vorzunehmen. Das Untersuchungsszenario sah zum einen die Visualisierung der Bewegungsprofile zweier Flurförderzeuge zur Materialbereitstellung in der Fertigung vor und zum anderen wurden für die Projektbearbeitung mehrfach Aufnahmen zur Annäherungsbestimmung (AP3.3/ 3.4) durchgeführt.

Im AP3 wurden zusammenfassend folgende Ergebnisse erzielt:

- Erstellung standardisiertes Versuchstemplate
- Mobile, leistungsfähige und konfigurierbare Applikation zur Aufnahme gesendeter Signaldatenpakete
- Funktionsfähige Datenverarbeitung innerhalb der mobilen Applikation und Ausgabe in Form .csv-Datei
- Funktionsfähiges Modul zur Datenaufbereitung mit geeigneten Verfahren zur Signaldatenaufbereitung (Rolling Mean, ARMA-Filter, Interpolationsfilter, Outlier Detection etc.) sowie Distanzkalkulation
- Erkenntnisgewinn über geeignete Parametereinstellung bei Datenaufnahme und Datenaufbereitung

(AP4) Prozesserkennung auf Basis aufgenommener  
Signalwerte

Für die Zielstellung einer automatisierten Zuordnung von definierten Prozessklassen zu den Signalstärkedaten mit dem Ziel einer hohen Übereinstimmung zur tatsächlich erfolgten Bewegung wurden in diesem AP geeignete Verfahren des Maschinellen Lernens insbesondere Klassifizierungsalgorithmen aufgegriffen, kombiniert und implementiert. Zudem erfolgte die

Ableitung und die Definition von Kategorien für produktionslogistische Prozesse auf Basis des Referenz-Szenarios Gabelstapler. Der wesentliche Entwicklungsschwerpunkt lag jedoch auf der Entwicklung, Umsetzung und Validierung eines Modells zur möglichst genauen Prozessvorhersage mit Hilfe von Machine Learning-Algorithmen.

#### AP4.1 Datenaufbereitung

Die Charakterisierung der notwendigen Prozessklassen für eine hinreichend genaue Prozesszuordnung ist abhängig von der betrachteten intralogistischen Prozesskette und kann je nach Anforderung und möglicher Granularität in einzelne Bausteine (Klassen) zerlegt werden. Im ersten Schritt wurde dies für das festgelegte Gabelstaplerrundfahrtproblem realisiert. Die Zerlegung der Prozesskette einer Gabelstaplerfahrt mit Materialumschlag wurde auf fünf Klassen reduziert: Aufsammeln, Ablegen, Fahrt ohne Ladung (Leerfahrt), Fahrt mit Ladung, Stillstand. Die einzelnen Klassen umfassen weitere detaillierte Unterprozesse. Hier kann z.B. genannt werden, dass ein Gabelstapler beim Aufnehmen von Ladung eine Positionierung der Gabel vor der Palette vollzieht. Andererseits kann die Prozesskette in weniger als fünf Klassen zerlegt werden: z.B. Fahrt mit Ladung, Fahrt ohne Ladung (Leerfahrt) und Stillstand. Die Zerlegung der Prozesskette unterliegt einer Vielzahl an Möglichkeiten und kann zudem auch von dem jeweiligen Untersuchungsszenario abhängig sein. Für die weitere Projektbearbeitung wurde sich für die o.g. Variante von fünf Prozessklassen entschieden, da eine Festlegung der Grenzen der definierten Klassen für die Variante gut umzusetzen war. Die Festlegung der Prozessgrenzen ist notwendig, da die Übergänge zwischen den Prozessen, bspw. bei „Aufsammeln“ zu „Fahrt mit Ladung“, klar abgegrenzt werden müssen, um Unschärfen bei der Prozesszuordnung zu reduzieren. Die Abgrenzung der Prozessklassen ist wie folgt definiert worden (siehe Tab. 6):

*Tabelle 6: Definition Prozessklassengrenzen*

<b>Stillstand</b>	Gabelstapler befindet sich nicht in Bewegung
<b>Aufsammeln</b>	Gabelstapler nimmt Material auf, inkl. Rangierprozess vor und nach Aufnahme
<b>Fahrt mit Ladung</b>	Gabelstapler hat Material auf Gabel und befindet sich in Fahrt, keine Rangierprozesse vor Lagerplatz
<b>Ablegen</b>	Gabelstapler legt Material ab, inkl. Rangierprozess vor und nach Aufnahme
<b>Fahrt ohne Ladung (Leerfahrt)</b>	Gabelstapler befindet sich ohne Material auf Gabel in Fahrt/ Bewegung

Die definierten Prozessklassen wurden, wie in AP3.2 bereits aufgezeigt, in der App zur Datenaufnahme hinterlegt.

#### AP4.2 Modellerstellung

Algorithmen aus dem Bereich des Machine Learnings bilden die Grundlage zur Erkennung von Mustern in großen Datensätzen. Die Mustererkennung wird im Besonderen bei einer automatisierten Zuordnung von Datensätzen zu vorgegebenen Datenklassen bzw. -kategorien angewendet. Es wird grundlegend nach unüberwachtem und überwachtem Lernen unterschieden. Für kategorisierte Daten, wie diese im Rahmen des Projektes vorliegen, bieten sich vor allem die Verfahren des überwachten Machine Learnings an. In der Literatur sind gängige Methoden im Bereich der Klassifikation zu finden. Um eine Eingrenzung und Auswahl geeigneter Lernalgorithmen zur automatischen Zuordnung von Prozessklassen zu Signalstärkedaten zu ermöglichen, wurden eine umfassende Recherche in der Forschungsliteratur und dem in der Scikit-Learn-Bibliothek zur Verfügung stehenden Schema (vgl. scikit-learn (2019)) durchgeführt sowie die Ergebnisse systematisiert. Für die Klassifizierung von Prozessdaten auf Basis von RSSI-Signalwerten fanden sich in der Literatur keine standardisierten



Klassifizierungsalgorithmen. Ein ähnliches Anwendungsfeld ist die Klassifizierung von menschlichen Bewegungsdaten (Motion Capture), welche bereits im industriellen Umfeld eingesetzt wird. Zur Mustererkennung von Bewegungsdaten kommen vorrangig folgende Algorithmen für eine Klassifizierung zur Anwendung: Support Vector Machine (SVM) (Feldhorst; ten Hompel (2016), Anguita et al. (2012)), Künstliche Neuronale Netze (kNN) (Pärkkä et al. (2006), Dutt et al. (2012) etc.), Random Forrest (RF) (Pavey et al. (2017), Feldhorst; ten Hompel (2016) etc.) sowie AdaBoost (Walse et al. (2016)). Ausgehend von diesen Erkenntnissen wurden für die Analyse-Methode die Klassifizierungsalgorithmen als Basis herangezogen und implementiert.

Für die Implementierung der Prozesszuordnung wurde das Software-Framework um ein Modul erweitert, in welchem die Klassifizierungsalgorithmen auf Basis der Python-basierten Scikit-Bibliothek hinterlegt wurden. Die Mustererkennungsverfahren aus dem Bereich des überwachten Lernens benötigen zum Anlernen der jeweiligen Algorithmen eine umfangreiche Datenbasis aus Trainings- und Testdaten. Trainingsdaten beinhalten alle wesentlichen Informationen, die für eine Mustererkennung bei dem Testdatenset notwendig sind. Je nach Komplexität der Datenbasis ist ein Anteil von 20% - 25% Trainingsdaten notwendig, um eine hinreichend genaue Mustererkennung bei den Testdaten zu gewährleisten. Somit bestand bei der Datenaufnahme die Anforderungen, die Signaldatenbasis je nach Untersuchungsszenario individuell zu generieren und dabei sicherzustellen, dass alle wesentlichen logistischen Prozessklassen beinhaltet sind. Wie bereits in AP3.2 und AP4.1 beschrieben, wurden die definierten Standardprozessklassen in der Applikation zur Datenaufnahme über Schaltflächen hinterlegt, um das Anlernen der jeweiligen Klassifizierungsalgorithmen mit spezifischen Trainingsdaten möglichst intuitiv zu ermöglichen. Die Ergänzung um individuelle Prozessklassen je nach Untersuchungsszenario wurde ebenfalls in der App vorgesehen.

Nachdem die vorbereitenden Maßnahmen abgeschlossen waren, wurden im experimentellen Rahmen erste Aufnahmen (fünf Versuche in unterschiedlicher Komplexität) von Trainings- und Testdatensätzen vorgenommen und diese zur Erprobung der Usability der Datenaufnahme, zur Prüfung der Eignung der ausgewählten Klassifizierungsalgorithmen sowie des entwickelten Software-Moduls eingesetzt. Die Eignung der Klassifizierungsalgorithmen wurde anhand der Klassifizierungsgenauigkeit (*CA* – classification accuracy) bewertet, welche ein Maß der korrekt klassifizierten Ergebnisse im Verhältnis zur Summe aller Ergebnisse darstellt (vgl. Brownlee 2014).

*Tabelle 7: Klassifizierungsergebnis Erprobungsstudie*

<b>Klassifizierungsgenauigkeit (CA)</b>					
<b>Versuch</b>	<b>SVM</b>	<b>RF</b>	<b>ABC</b>	<b>kNN</b>	<b>Durchschnitt</b>
<b>V1</b>	88,1%	90,1%	88,4%	91,3%	<b>89,5%</b>
<b>V2</b>	81,3%	77,3%	76,5%	79,2%	<b>78,6%</b>
<b>V3</b>	86,9%	83,4%	43,5%	85,5%	<b>74,8%</b>
<b>V4</b>	70,7%	66,6%	34,5%	77,0%	<b>62,2%</b>
<b>V5</b>	68,5%	59,3%	49,9%	70,0%	<b>62,9%</b>
<b>Durchschnitt</b>	<b>79,1%</b>	<b>75,3%</b>	<b>58,6%</b>	<b>80,6%</b>	

In der Erprobungsstudie konnte festgestellt werden, dass die implementierten Klassifizierungsalgorithmen gute bis sehr gute Ergebnisse hinsichtlich der Klassifizierungsgenauigkeit lieferten (siehe Tab. 7). Dies bedeutete für die weiteren Untersuchungen, dass alle implementierten Klassifizierungsalgorithmen für die weiterführenden

Entwicklungen übernommen wurden. Des Weiteren steht die Leistungsfähigkeit der einzelnen Klassifizierungsalgorithmen in Zusammenhang mit der Komplexität des Untersuchungsszenarios und der Datenbasis, was je nach Datengrundlage dazu führen kann, dass verschiedene Klassifizierer bei unterschiedlichen Untersuchungsszenarios zu favorisieren sind.

Eine detaillierte Analyse der Klassifizierungsergebnisse ergab ergänzend dazu, dass die Klassifizierungsalgorithmen v.a. in den Übergangsbereichen zwischen den Prozessklassen Schwächen bei der korrekten Zuordnung aufwiesen (siehe Abb. 12).

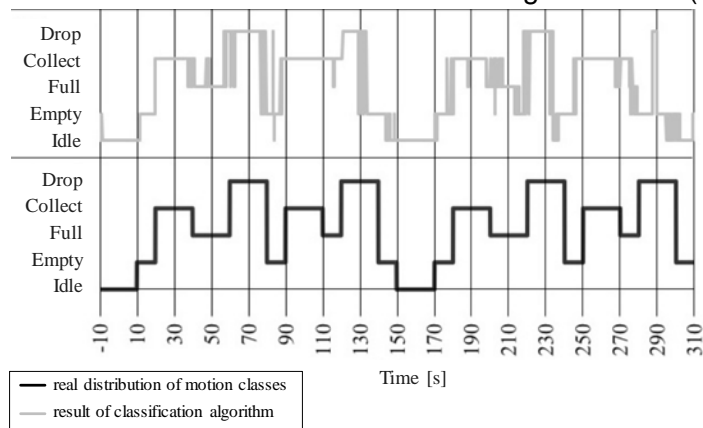


Abbildung 12: Darstellung fehlerhafte Zuordnung Prozessklassen

Aufgrund der fehlerhaften Zuordnung der Prozessklassen wurde ein Konzept zur Verbesserung der Prozessvorhersage für die ausgewählten Klassifizierungsverfahren abgeleitet. Dieses sah die folgenden beiden wesentlichen Schritte vor:

1. Entwicklung eines retrospektiven regelbasierten Prüfalgorithmus zur Reduzierung fehlerhafter Prozesszuordnungen
2. Einbindung einer automatisierten Optimierung der Hyperparameter der Klassifizierungsalgorithmen

Im ersten Schritt wurde die Implementierung von Regelsets, welche nach den eigentlichen Klassifizierungsalgorithmen das Klassifizierungsergebnis hinsichtlich logischer Klassenzuordnung, erreichter Vorhersagewahrscheinlichkeit sowie weiteren Prüfparametern unetrsuchen und ggf. anpassen, umgesetzt. Als Ausgangsbasis wurde eine Reihenfolgematrix erstellt, welche Kombinationsmöglichkeiten von Prozessklassen definiert und somit die Logikbeziehungen zwischen den Klassen vorgibt (siehe Abbildung 13).

	Stillstand	Aufsammeln	Fahrt mit Ladung	Ablegen	Fahrt ohne Ladung
Stillstand	x	x	x	x	x
Aufsammeln	x	x	x	-	-
Fahrt mit Ladung	x	-	x	-	-
Ablegen	x	x	-	-	x
Fahrt ohne Ladung	x	x	-	-	-

Abbildung 13: Reihenfolgebeziehung Prozessklassen

Des Weiteren wurde ebenfalls geprüft, welchen Einfluss die Größe der Datenbasis auf die Genauigkeit der Klassifizierungsalgorithmen sowie die Leistungsfähigkeit der Regelsets hat. Es hat sich gezeigt, dass die Datenbasis mindestens 12.000 Datensätze umfassen sollte, um aussagekräftige Ergebnisse generieren zu können. Die notwendige Größe der Datensätze steigt

mit der Komplexität des Untersuchungsszenarios an. Mit dem entwickelten Algorithmus konnten sehr gute Ergebnisse erzielt werden. Im Durchschnitt konnten die Klassifizierungsergebnisse hinsichtlich des  $CA$ -Wertes um 30% verbessert werden. Die einzelnen Regelsets wurden sukzessive erarbeitet, an mehreren Versuchsdatensätzen mit unterschiedlicher Komplexität geprüft sowie in das entsprechende Software-Modul implementiert.

Neben den Inputdaten haben vor allem die Parametersettings der verwendeten Mustererkennungsverfahren einen wesentlichen Einfluss auf die Klassifizierungsgüte. Jeder Klassifizierungsalgorithmus gibt ein eigenes Parameterset vor, welches je nach Wert der einzelnen Parameter die Leistungsfähigkeit der Klassifikatoren stark beeinflusst. Detaillierte Analysen auf Basis durchgeführter experimenteller Studien haben bspw. für die SVM ohne Einsatz von Kernels einen deutlichen Leistungsabfall gezeigt, ebenso beeinflussen einige Parameterkombinationen mit Kernel die Genauigkeit des Klassifizierers negativ (siehe Tabelle 8).

*Tabelle 8: CA bei variierenden Parametereinstellungen*

Klassifizierer	SVM			
	ohne Kernel- Funktion	Linear		
Solver		L-BFGS-B	Logistic	
	SGD		Adam	
<b>V1</b>	0,63	0,90	0,70	0,92
<b>V2</b>	0,39	0,73	0,68	0,68
<b>V3</b>	0,52	0,86	0,76	0,87
<b>V4</b>	0,46	0,61	0,73	0,66
<b>V5</b>	0,38	0,70	0,65	0,66
<b>Durchschnitt</b>	<b>0,48</b>	<b>0,76</b>	<b>0,70</b>	<b>0,76</b>

Auf Basis dieser Untersuchungsergebnisse wurde deshalb im zweiten Schritt eine Szenario-basierte Optimierung der Hyperparameter der jeweiligen Klassifizierungsalgorithmen umgesetzt. Ziel war die Auswahl, Automatisierung und Evaluierung von geeigneten Parametrierungsverfahren speziell für die Hyperparameter von Klassifizierungsalgorithmen. Die Hyperparameter unterscheiden sich von den Modellparametern in der Hinsicht, dass diese vor dem Training des Klassifikators vom Nutzer bzw. einem Algorithmus angepasst werden und während des Trainings konstant bleiben. Auf Basis einer Literaturrecherche wurden für die Klassifizierungsalgorithmen geeignete Methoden zur Hyperparameteroptimierung (HPO) herausgesucht. Zur Umsetzung wurden folgende die Methoden herangezogen:

- Rastersuche (RS)
- Zufällige Suche (ZS)
- Genetische Algorithmen (GA)
- Partikelschwarmoptimierung (PSO).

Die Implementierung der Optimierungsmethoden ebenso wie deren automatisierten Ablauf wurde im Framework realisiert. Auf Basis der Verbesserung der Klassifizierungsgenauigkeit  $\Delta CA_{HPO}$  bei Verwendung der optimierten Hyperparameter im Vergleich zu verwendeten Standard-Hyperparameterwerten sowie der benötigten Rechendauer erfolgte die Bewertung der Leistungsfähigkeit der einzelnen Methoden zur Hyperparameteroptimierung. In Tabelle 9 sind die Ergebnisse beispielhaft für den Klassifizierer SVM dargestellt. Die generierten Ergebnisse zeigten deutlich, dass eine Optimierung der Hyperparameter von Klassifizierungsalgorithmen die Leistungsfähigkeit erhöht. Die höchste Verbesserung der  $CA$  liegt bei über 28 % im Vergleich zur  $CA$  bei der Verwendung von Standardparametern.

Tabelle 9: Verbesserung CA durch Hyperparameteroptimierung (HPO)

SVM	CA [%] Versuch 1	$\Delta CA_{HPO}$	CA [%] Versuch 2	$\Delta CA_{HPO}$	CA [%] Versuch 3	$\Delta CA_{HPO}$	$\emptyset$ Verbesserung zu Standard
<b>Standard</b>	71,3		46,3		35,0		
<b>GS</b>	77,7	6,4	56,0	9,7	59,9	24,9	13,7
<b>RS</b>	78,1	6,8	55,6	9,3	59,0	24,0	13,4
<b>GA</b>	73,3	2,0	56,5	10,2	60,6	25,6	12,6
<b>PSO</b>	<b>78,2</b>	6,9	<b>60,8</b>	14,5	<b>60,8</b>	25,8	<b>15,7</b>

Mit den Optimierungsmethoden PSO und GA ergaben sich durchschnittlich die höchsten Potentiale zur Verbesserung der CA. Für die Implementierung wurden deshalb die Optimierungsalgorithmen dieser beiden Verfahren übernommen.

Wie bereits erwähnt, wurde zur Eignungsprüfung der ausgewählten Klassifizierungsalgorithmen eine Erprobungsstudie mit mehreren Versuchen durchgeführt, um die Generierung der notwendigen Datenbasis aus Trainings- und Testdaten zu gewährleisten. In diesem Zuge wurde auch die Handhabbarkeit der Prozesseingabe via App bei der Datenaufnahme von den Entwicklern getestet. Für jeden weiteren Entwicklungsschritt wurden umfangreiche Experimente zur Datengenerierung und zur Funktionsprüfung der einzelnen Software-Pakete, insbesondere der HPO und der Anwendung des Regelsetalgorithmus, durchgeführt, dokumentiert und diese nach wissenschaftlichen Maßstäben ausgewertet. Ergänzend dazu wurden wie in AP3 beschrieben bei Unternehmen des PA Feldversuche durchgeführt, die einen besonderen Fokus auf die Usability der Aufnahme von Trainingsdaten mit gleichzeitiger Prozesshinterlegung legte. Im Rahmen der Feldversuche führten die Mitarbeiter der KMU, insbesondere die Gabelstaplerfahrer, die Datenaufnahme selbst durch. Neben dem normalen Tätigkeitsumfang kam für diese ungeübten Anwender noch die Bedienung des Smartphones hinzu. Die Mitarbeiter konnten somit ein umfassendes Feedback zu den definierten Prozessgrenzen sowie zur Prozesshinterlegung via Smartphone geben. Die definierten Prozessgrenzen erschienen allen Anwendern logisch und gut formuliert, jedoch lag die eigentliche Herausforderung in der händischen Prozesseingabe, welche zu verzögerten Eingaben nach Prozesswechsel führte. Diese Unschärfen konnten nachgelagert auch in der Datenbasis nachgewiesen werden. Der entwickelte Regelalgorithmus zur Behebung fehlerhaft klassifizierter Prozesse konnte diese in einem ausreichenden Maß reduzieren, sodass nachgelagert kein negativer Einfluss auf die Prozessvorhersage bestand. Für weitere Feldversuche wurde die Aufnahme von Trainingsdaten durch einen Projektmitarbeiter realisiert, v.a. auch um die hohe Ablenkung durch die Smartphone-Bedienung und somit ein etwaiges Unfallrisiko bei den Gabelstaplerfahrern zu vermeiden.

Im AP4 wurden zusammenfassend folgende Ergebnisse erzielt:

- Definition Prozessklassen für Referenz-Prozesskette (Gabelstapler)
- Auswahl geeigneter Klassifizierungsmethoden
- Verbesserung der Klassifizierungsgenauigkeit für vordefinierte Prozesse durch entwickelten Prüfalgorithmus sowie automatisierter Hyperparameteroptimierung
- Anbindung funktionsfähiges Software-Modul zur Prozesserkennung an Software-Framework und Datenbank

#### (AP5) Entwicklung und Test Demonstrator

Die Zielstellung des fünften Arbeitspaketes ist die Modellvalidierung mit Hilfe eines geeigneten Demonstrator-Tools. Die fokussierte mobile sowie Java-basierte Applikation wurde nur für die

Datenaufnahme und Systeminitialisierung umgesetzt. Da die übrigen Software-Module sowie die Anbindung der Datenbank Python-basiert entwickelt wurden, erfolgte die Anbindung des Demonstrator-Moduls für die Datenvisualisierung sowie -auswertung ebenfalls Python-basiert. Die Entwicklungsschritte wurden zudem parallel zu den bereits erläuterten AP 2, 3 und 4 umgesetzt.

#### AP5.1 Entscheidungsgestütztes Analysedesign

In AP 1.1 und AP 3.1 wurde verschiedene Vorlagen zur Versuchsdurchführung in Abhängigkeit des Untersuchungsszenarios erarbeitet und ausführlich erläutert.

#### AP5.2 GUI zur Primärinitialisierung und Systemkonfiguration

An dieser Stelle ist auf die ausführliche Beschreibung der App-Entwicklung zur Datenaufnahme in AP3.2 sowie in der Anlage verweisen. Die entwickelte App beinhaltet alle geplanten Funktionalitäten zur Konfiguration und Primärinitialisierung des Analysesystems.

Die Markierung von vorhandenen Störgrößen, wie bspw. Maschinen konnte ebenfalls erfolgreich implementiert werden, jedoch nicht in der mobilen Applikation, sondern in der GUI zur Visualisierung der Bewegungsprofile (siehe AP3.4).

#### AP5.3 Datenspeicherung

Wie in AP2 dargelegt, erfolgte die Erstellung eines Datenbankmodells mit Hilfe der dokumentbasierten NoSQL-Datenbank MongoDB. Die frei verfügbare Open Source Datenbank konnte zudem erfolgreich an das Software-Framework angebunden werden. Lediglich die via App aufgenommenen Signalstärkedaten müssen manuell eingelesen werden, alle weiteren Module ziehen und schreiben die Daten automatisiert aus bzw. in die Datenbank.

#### AP5.4 Modell-Analyseeinheit

Die wesentliche Zielstellung für die Entwicklung der Analyseeinheit war das Schaffen eines Moduls zur übersichtlichen Darstellung von Kenngrößen, Auswertungscharts und den wesentlichen Informationen zum Untersuchungsszenario bzw. zum Rahmen der Datenaufnahme. Die Daten und Informationen werden dafür im Ursprung aus verschiedenen Quellen zur Verfügung gestellt und über die Datenbank gebündelt abgerufen. Dafür ist im entwickelten Analyser-Modul zu Beginn eine Anbindung an die Datenbank geschaffen wurden, die die benötigten Datensätze aus der Datenbank zieht.

Die Analyseergebnisse umfassen einen Report zum Untersuchungsszenario, welcher folgende Informationen enthält /siehe Abb. 14):

- Layoutdatei
- Informationsübersicht Datenaufnahme, z.B.:
  - Ort Datenaufnahme
  - Start/ Ende der Aufnahme
  - Anzahl Beacon erkannt/ initialisiert
  - Anzahl Zeitstempel
  - Anzahl Messungen
  - Anzahl Fixpunkte

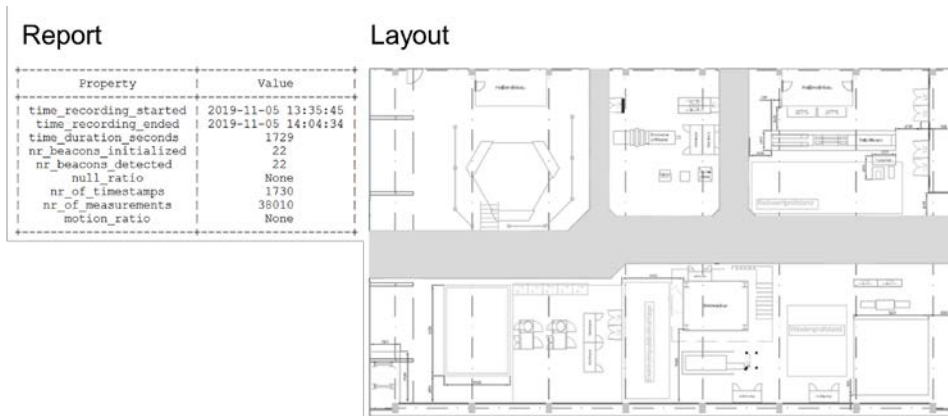


Abbildung 14: Darstellung Informationen Datenaufnahme in Analyser-Modul

Des Weiteren wurden diverse logistische Kenngrößen integriert und berechnet, welche die Grundlage für die Visualisierung von Prozess-Ganttcharts sowie Prozessklassendiagramme (bspw. siehe Abb. 15) darstellen.

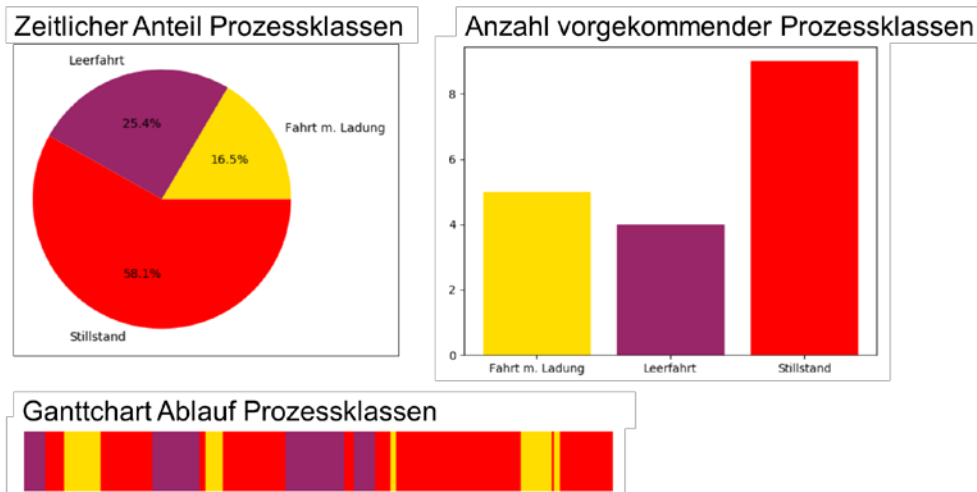


Abbildung 15: Visualisierung Prozessablauf und Kennzahlenanalyse

Der Analyser-Modul ist über die graphische Benutzeroberfläche (GUI) des Visualisierungsmoduls (AP5.5) aufrufbar. Die Analyseergebnisse werden als .html-Dokument zusammenfassend dargestellt.

### AP5.5 Visualisierung

Ein wesentlicher Bestandteil einer fundierten Prozessanalyse ist die Ergebnisvisualisierung. Nur so ist es möglich, die untersuchten Prozesse, aufgenommenen Daten sowie deren zielgerichtete Aufbereitung für die Analyse und Auswertung aber auch für die Ergebnispräsentation darstellbar zu machen.

Die zu entwickelnde graphische Oberfläche sollte vorrangig dazu dienen, die generierten Distanz- und Positionsdaten auf die zugrundeliegenden Layoutdaten zu übertragen und somit die Visualisierung von Bewegungsprofilen zu ermöglichen. Zur Umsetzung der graphischen Oberfläche wurde ein weiteres Software-Modul im Framework integriert, welches ebenfalls Python-basiert (GUI Toolkit PyQT 5) arbeitet. Die implementierten Funktionalitäten umfassen folgende Möglichkeiten zur Bearbeitung von Datensätzen sowie zur Bearbeitung der Visualisierung von Positionsprofilen:

- Datensuche
- Datenimport/ Datensätze löschen

- Anwendung von Glättungs- sowie Interpolationsfilter mit Parameterauswahl und Distanzkalkulation (siehe Abb. 16)

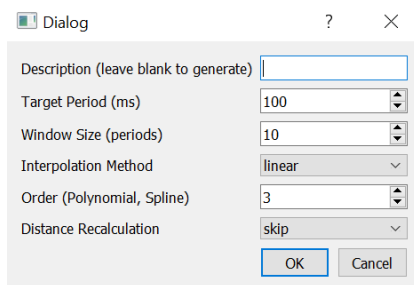


Abbildung 16: GUI-Dialog zur Auswahl Glättungs- und Filtermethoden

- Datenklassifizierung mit Auswahl Klassifizierer und deren Parameter
- Bearbeitung Layoutdatei vor Visualisierung Bewegungsprofile: Einfügen „Restricted Areas“
- Anzeige Bewegungsprofile auf Layout (Bewegungslinien oder –punktwolken, farbliche Hervorhebung verschiedener Prozessklassen im Bewegungsprofil)
- Generieren von Head Maps
- Anzeige von Signalverläufen je Beacon
- Start des Analyser-Moduls

Die Ausgabe der generierten Bewegungsprofile (sowohl mit/ ohne Head Map oder Restricted Areas) erfolgt aus dem GUI-Skript in das Webinterface Leaflet, welches die Möglichkeiten bietet die Visualisierungen leicht gestalterisch zu variieren (Anzahl angezeigter Elemente, z.B. Beacons oder Variation der Farbgebung) sowie an die Positionspunkte Daten-Popups anzubinden. Darstellungsbeispielen zeigen die Abbildungen 17 und 18. Die Speicherung der generierten Visualisierungen erfolgt ebenfalls über das Leaflet-Interface.

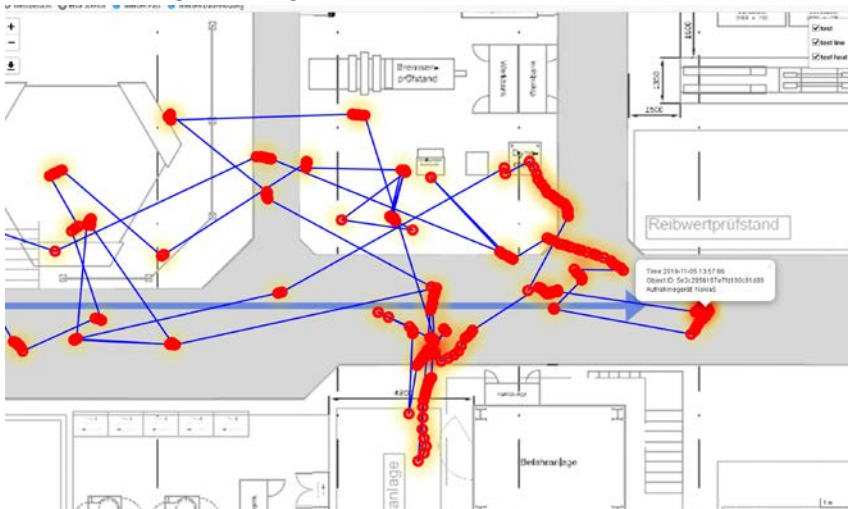


Abbildung 17: aus GUI generiertes Bewegungsprofile



Abbildung 18: aus GUI generierte Head Map mit Restricted Areas

### AP5.6 Verifikation und Validierung

Die Erprobung der entwickelten GUI erfolgte bereits permanent während der Entwicklung. So wurden Datensätze genutzt, die während der gesamten Projektlaufzeit im Büromaßstab, in universitären Versuchshallen oder Turnhallen im Rahmen der Entwicklungsarbeit der anderen Arbeitspakete generiert wurden sind. So war eine umfangreiche Datengrundlage in hoher Variation vorhanden, die eine umfangreiche Erprobung und Bewertung der Funktionalitäten ermöglichte.

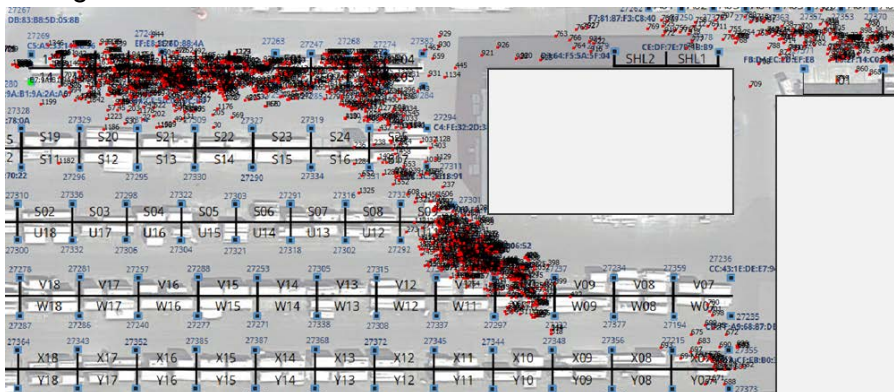


Abbildung 19: GUI-basiertes Bewegungsprofil Gabelstapler Kommissionierbereich

Ergänzend dazu wurde das GUI im Rahmen der Feldversuche bei Unternehmen des PA eingesetzt, um die Datenbasis aufzubereiten, zu analysieren und die Ergebnisse zu visualisieren. In Abbildung 19 ist beispielhaft das Bewegungsprofil eines Gabelstaplers im Kommissionierbereich eines der Unternehmen dargestellt. Hier wurden zudem zur Optimierung der Positionsbestimmung Restricted Areas eingebunden.

### (AP6) Dokumentation und Veröffentlichung

Zur Dokumentation der generierten Ergebnisse wurden bereits sowohl wissenschaftliche Vorträge gehalten als auch eine Publikation verfasst. Eine weitere Veröffentlichung im Rahmen eines wissenschaftlichen Beitrags für eine Fachzeitschrift ist in Bearbeitung. Die Erstellung eines Leitfadens zur Verwendung der Methode sowie die technische Dokumentation der einzelnen Module des entwickelten Frameworks sind während der Projektlaufzeit erfolgt und bereits abgeschlossen. Des Weiteren befinden sich weitere Transfermaßnahmen in Planung und Vorbereitung sowie wurden bereits umgesetzt (siehe Plan zum Ergebnistransfer). Der entwickelte



Software-Demonstrator wird auf Anfrage bei der Forschungsstelle allen Interessierten zur Verfügung gestellt.

## 2. Verwendung der Zuwendung

TL:

- Im Berichtszeitraum wurden zur Durchführung der Arbeiten 24 Monate wissenschaftlich-technisches Personal (Einzelansatz A.1 des Finanzierungsplans) eingesetzt.
- Es wurden fünf studentische Hilfskräfte in die Bearbeitung mit einbezogen und sechs Abschlussarbeiten sowie vier Studienarbeiten verfasst.
- Das Gerät (Einzelansatz B des Finanzierungsplans) „Versuchsanlage zur temporären intralogistischen Analyse“ konnte nicht im Rahmen der „Ausgaben für die Gerätebeschaffung“ beschafft werden, da aufgrund stark gefallener Preise der Einzelbeschaffungswert von mehr als 2.500,- € (einschließlich Umsatzsteuer) nicht erreicht werden konnte. Die Versuchsanlage wurde daher aus der „Pauschale für Sonstige Ausgaben“ finanziert. Stattdessen konnte ein QuantumX Universalmessverstärker beschafft werden, der es ermöglichte über die forcierten Projektergebnisse hinaus, eine hochpräzise Signalstärkedatenbasis zu generieren, um zum einen die Leistungsfähigkeit hinsichtlich Signaldatenverarbeitung der für die Projektbearbeitung verwendeten Empfangsgeräte zu bewerten und somit eine Auswahl an geeigneten Empfangsgeräten für die optimale Datenaufnahme zu gewährleisten. Zum anderen konnten mit Hilfe des Universalmessverstärkers ergänzende Sensordaten eingebunden werden, um die Plausibilitätsprüfung der durch die Software-Module abgeleiteten Daten durchzuführen. Somit konnte das Projektergebnis positiv beeinflusst werden.
- Für Leistungen Dritter (Einzelansatz C des Finanzierungsplans) wurden keine Ausgaben geplant.

## 3. Notwendigkeit und Angemessenheit der geleisteten Arbeit

Die durchgeführten Arbeiten leisten einen angemessenen Beitrag zum Forschungsvorhaben der Entwicklung eines universellen Mess- und Auswertungssystems zur zeitlich begrenzten Analyse logistischer Systeme. Dabei konnte die Komplexität der durchgeführten Arbeiten mit Hilfe von mehreren Studien- und Abschlussarbeiten auf ein höheres Niveau gebracht und im Bereich der Datenlückenkompensation sogar noch übertroffen werden. Die durchgeführten Arbeiten waren in Inhalt und Umfang ein notwendiger Teilschritt zur Erreichung des geplanten Forschungsziels. Die Ergebnisse des Projektes entsprechen den Erwartungen des Arbeitsablaufs. Zum Erreichen der Zielstellung wurden Mitarbeiter laut Plan beschäftigt.

#### 4. Darstellung des wissenschaftlich-technischen und wirtschaftlichen Nutzens der erzielten Ergebnisse insbesondere für KMU sowie ihres innovativen Beitrags und ihrer industriellen Anwendungsmöglichkeiten

Im Rahmen des Forschungsvorhabens konnte eine KMU-gerechte Methode zur gezielten sowie temporären Analyse von intralogistischen Prozessen entwickelt werden, welche zudem durch den Einsatz preiswerter, wiederverwendbarer Sender- und Empfängertechnologie in Verbindung mit geringem Installationsaufwand die Anwendbarkeit für KMU auch ohne Vorkenntnisse ermöglicht. Der geringe Installationsaufwand wird durch eine grafikbasierte, intuitive Drag-and-Drop-Lösung für die Systemprimärinitialisierung und durch eine Hinterlegung von Szenario-abhängigen Eingangsgrößen direkt via Smartphone erreicht. Das entwickelte modulare Software-Framework ermöglicht eine gezielte Datenaufbereitung, -verarbeitung und -analyse. Sowohl die Zuordnung der definierten Prozessklassen zu den Signalstärkedatensets als auch die Umsetzung eines Positionierungsalgorithmus ohne Notwendigkeit starrer Senderfixierung konnte im Software-Framework implementiert, erfolgreich getestet und an realen Fallbeispielen angewendet werden. Die Visualisierung der abgeleiteten Bewegungsprofile und der in diesem Zusammenhang durchgeführten Prozesse sind ebenso Bestandteil wie eine umfassende Auswertungseinheit mit der Möglichkeit zur Darstellung von Kenngrößen, Gantt-Charts sowie Klassendiagrammen.

Unternehmen und Anwender werden somit mit diesen Methoden befähigt, je nach Bedarf selbstständig in zeitlichen Abständen und dem Untersuchungsziel angepasst, Daten aufzunehmen und auszuwerten. Die in intralogistischen Prozessen häufig auftretenden ineffizienten Abläufe sind mit der Analyse-Methode auch ohne hohen monetären und zeitlichen Aufwand quantifizierbar. Dies versetzt KMU in die Lage, selbstständig ihre intralogistischen Prozesse zu analysieren, Verschwendung zu erkennen sowie Maßnahmen zu deren Vermeidung abzuleiten. Die flexible Einsetzbarkeit sowie die temporäre Datenaufnahme unterstützen zudem kontinuierliche „Lean-Projekte“ (KVP) und tragen somit ebenfalls zur Sicherung der Wettbewerbsfähigkeit bei. Darüber hinaus können mit dem entwickelnden System strategisch die Potenziale bzw. Schwerpunkte des erforderlichen durchgängigen Digitalisierungsprozesses besser bewertet werden und Investitionspriorisierungen werden transparenter.

Die zentrale Innovation besteht bei der entwickelten ganzheitlichen Prozessanalyse-Methode darin, dass mit verschiedenen Methoden zur temporären Analyse intralogistischer Systeme bei moderaten Investitions-, Installations- und Betriebskosten eine äquivalent hohe Analysequalität und -detaillierung im Vergleich zu bestehenden Systemen der digitalen Datenerfassung geschaffen wurde. Basis bildet die kostengünstige Sender- (Beacon) und Empfängertechnologie (Smartphone). Mit deren Hilfe eine ereignisbasierte Lokalisierung ausschließlich auf Basis von BLE-Signaldaten (RSSI) und Positionsdaten realisiert werden konnte. Dabei erfolgt kein Eingriff in laufende Prozesse. Zudem kann das entwickelte benutzerfreundliche Installations- und Anwendungssystem flexibel, intuitiv und ohne Anbindung an unternehmensinterne Software betrieben werden. Das Innovationspaket setzt sich somit aus folgenden Entwicklungen zusammen:

- Realisierung Systemselbstkonfiguration zur Reduzierung von Senderfixpunkten und einer benutzerfreundlichen App zur Systeminitialisierung sowie automatisierten Datenaufnahme
- automatisierte Zuordnung von Signalstärkedaten zu vordefinierten intralogistischen Prozessen

retrospektive Ableitung von Positionsprofilen Materialflussobjekte auf Basis relativer Signalstärkedaten sowie Layoutdaten

## 5. Plan zum Ergebnistransfer in die Wirtschaft

Folgende Transfermaßnahmen wurden im Berichtszeitraum durchgeführt:

Während der gesamten Projektlaufzeit wurde der intensive Kontakt mit Partnern des Projektbegleitenden Ausschusses gepflegt und weitergeführt. Dabei wurde zum einen intensive Diskussionen zu unternehmensinternen Untersuchungsszenarien, allgemeinen Einsatzmöglichkeiten sowie während der Entwicklung und Erprobung zu Optimierungs- und Erweiterungspotentialen der Analyse-Methode geführt, um stets auf Vorstellungen und Bedürfnisse aus der Wirtschaft eingehen zu können. Zum anderen konnte durch den intensiven Austausch eine Validierung der Entwicklung durch mehrere Realanwendungen umgesetzt und werden. Besonders enger Kontakt wurde zu den Unternehmen Porenbetonwerk Europor GmbH, WaCo GERÄTECHNIK GmbH und Elbe Flugzeugwerke GmbH gehalten. Hier kann sich zukünftig eine weitere Forschungszusammenarbeit entwickeln.

Folgende Maßnahmen werden nach dem Berichtszeitraum durchgeführt:

Auf Basis der Entwicklungsergebnisse ist es vorgesehen gemeinsam mit der Saxony Media Solutions GmbH, ebenfalls ein Mitglied des PA, ein praxisnahes Projekt zu realisieren. Zielstellung ist es, die Erkenntnisse aus den Annäherungsversuchen hinsichtlich Dämpfungseffekten sowie Störquellen im Produktions- und Logistikumfeld in ein Anwendungssystem zur optimalen Sender-Empfänger-Positionierung einfließen zu lassen.

Transfermaßnahmen während der Projektlaufzeit:

Maßnahme	Ziel	Ort / Rahmen	Datum / Zeitraum
Information der Unternehmen des Projektbegl. Ausschusses (PA)	Abstimmung Arbeitsstand & Vorgehensweise ausführlicher Ergebnistransfer	PA-Sitzung	11.2018
	ausführlicher Ergebnistransfer		10.2019 02.2020
Enge kooperative Zusammenarbeit mit Unternehmen des Projektbegl. Ausschusses (PA)	Feldversuche und Anwendung der Methodik im industriellen Umfeld	PA-Unternehmen	Mehrere Vororttermine zur Abstimmung und Vorbesprechung 05.2019 – 02.2020
	Erkenntnisgewinn über Funktions-fähigkeit und Entwicklungsstand der Methodik PA-Unternehmen		
Ansprache weiterer interessierter Unternehmen	Gewinnung zur Mitarbeit im PA, Problemsensibilisierung	Stuttgart / Logimat	03.2018
	weitere Verbreitung der Technologie in der Wirtschaft, besonders der Intralogistik	Dresden / 1. Symposium Flurförderzeuge	09.2018
	Einbindung interessierter Unternehmen, FV und Verbände	Chemnitz / Trans <sup>3</sup> Net.Show: Digitalisation of Industry	09.2018

Internetpräsenz	Wissenstransfer		Q4 2018
Wissenschaftliche bzw. industriennahe Vorträge	Wissenstransfer an interessiertes Fachpublikum & Diskussionen mit Fachexperten	Paris / ICIME	04.2019
		Rostock / 15. WGTL-Fachkolloquium	09.2019
	Wissenstransfer, Verifikation & Validierung	Unternehmen des PA	2019/ 2020
Wissenschaftliche Publikationen in Fachzeitschriften/ Dissertationen	Verbreitung der Ergebnisse an interessierte Fachleute bzw. Unternehmen	Veröffentlichung in Fachzeitschriften	12.2019 Tagungsbandbeitrag zum 15. WGTL-Fachkolloquium
	Vermitteln von Kenntnissen und Heranführen an eigenständiges wissenschaftliches Arbeiten	Studien-, Diplomarbeiten, Dissertation	08.2018 Diplomarbeit: Klassifizierung logistischer Prozesse auf Basis von Signalstärke-Distanz Messungen mit der Beacon-Technologie
12.2018 Diplomarbeit: Entwicklung eines modularen Versuchsframeworks zur Analyse produktionslogistischer Fragestellungen mit Hilfe der Beacon-Technologie			
02.2019 Diplomarbeit: Untersuchung von Dämpfungsfaktoren in Produktionshallen auf die Empfangsempfindlichkeit von Beacon-Signalen			
03.2019 Studienarbeit: Verbesserung der Klassifikationsgenauigkeit logistischer Prozesse			
11.2019 Diplomarbeit: Parametrierung von Klassifikatoren zur Prozessidentifikation im logistischen Produktionsumfeld auf Basis von Signalstärke-Distanz-Daten			
02.2020 Diplomarbeit: Unternehmensbezogene Bewertung und Optimierung logistischer Prozesse: Konzeption und Entwicklung eines KMU-spezifischen Reifegradmodells			
Abschlussbericht	Wissenstransfer		Q2 2020

## Transfermaßnahmen nach der Projektlaufzeit:

Maßnahme	Ziel	Ort / Rahmen	Datum / Zeitraum
Beratung von Unternehmen	Beratung von Unternehmen bei der	Interessierte KMU	ab 05.2020

	Steuerung temporärer Produktionsnetze		
Wissenschaftliche Publikation in Fachzeitschriften	Verbreitung der Ergebnisse an interessierte Fachleute	Journal-Paper	Q3 2020
Fortführungsprojekt	Kooperationsprojekt mit Unternehmen aus PA zur Umsetzung eines Werkzeug zur optimalen Sender-Empfänger-Positionierung	Kooperation KMU – TUD TL	ab Ende 2020
Übernahme der Ergebnisse in die akademische Lehre	Lehrveranstaltungen TL: „Produktion und Logistik“ und „Projektmanagement“		ab Q3/2020

## Quellen

Android Beacon Library (2019): Distance Calculation. <https://altbeacon.github.io/android-beacon-library/distance-calculations.html>, abgerufen am: 10.02.2019.

Anguita, D., Ghio, A., Oneto, L., Parra, X. & Reyes-Ortiz, J. L. (2012): Human Activity Recognition on Smartphones Using a Multiclass Hardware-Friendly Support Vector Machine. In: Ambient assisted living and home care. [4th international workshop IWAAL 2012, 3.-5.12.2012, Vitoria-Gasteiz, Spanien]. Berlin: Springer, ISBN: 9783642353949, S. 216-223.

Brownlee, J. (2014): Classification Accuracy is Not Enough: More Performance Measures You Can Use. <https://machinelearningmastery.com/classification-accuracy-is-not-enough-more-performance-measures-you-can-use/>.

Dutt, V., Chaudhry, V. & Khan, I. (2012): Pattern Recognition: an Overview. In: American Journal of Intelligent Systems, Vol. 2, Nr. 1, S. 23–27.

Evennou, F.; Marx, F. (2006): Advanced Integration of WiFi and Inertial Navigation Systems for Indoor Mobile Positioning. In: EURASIP Journal on Advances in Signal Processing, DOI: 10.1155/ASP/2006/86706.

Feldhorst, S. & ten Hompel, M.: Bewegungsklassifikation mithilfe mobiler Sensoren. In: Logistics Journal. Proceedings. Online. 2016, DOI: 10.2195/lj\_Proc\_feldhorst\_de\_201605\_01.

Pavey, T. G., Gilson, N. D., Gomersall, S. R., Clark, B. & Trost, S. G. (2017): Field evaluation of a random forest activity classifier for wrist-worn accelerometer data. In: Journal of science and medicine in sport, Vol. 20, Nr. 1, S. 75–80.

Pärkkä, J., Ermes, M., Korpipää, P., Mäntylä, J., Peltola, J. & Korhonen, I. (2006): Activity Classification Using Realistic Data From Wearable Sensors. In: IEEE Transactions on Information Technology in Biomedicine. Vol. 10, Nr. 1, S. 119–128

scikit-learn (2019): Documentation of scikit-learn. [https://scikit-learn.org/stable/supervised\\_learning.html](https://scikit-learn.org/stable/supervised_learning.html), 2018. Abgerufen am: 01.04.2019.

Vaughan, R.; Andersen, J. B. (2003): Channels, Propagation and Antennas for Mobile Communications. In: IEEE Antennas and Propagation Magazine, Vol. 45 , Nr. 4 , DOI: 10.1109/MAP.2003.1241316.

Walse, K. H., Dharaskar, R. V. & Thakare, V. M. (2016): A Study of Human Activity Recognition using Adaboost Classifier on WISDM Dataset. In: The IIOAB Journal, Vol. 7, Nr. 2, S. 68–76.