

Bachelorarbeit

Prädiktion der Greifpositionen bei bekannten Zielen anhand der initialen Handtrajektorie

Autor

Burak Bayram

Betreuer

Nico Höllerich

Prüfer

Prof. Dr. Dominik Henrich

27. September 2024

Lehrstuhl für Angewandte Informatik III

Robotik und Eingebettete Systeme

Universität Bayreuth

Zusammenfassung

In der Mensch-Roboter-Zusammenarbeit ist eine korrekte Vorhersage menschlicher Handbewegungen essenziell, um eine sichere sowie effiziente Interaktion zu gewährleisten. Der verfolgte Ansatz in dieser Arbeit ermöglicht mithilfe von wenigen Messdaten eine frühzeitige Erkennung von bekannten Zielen, die von der Hand am wahrscheinlichsten angestrebt werden. Das Ziel besteht darin, das Verfahren zu analysieren und in Python zu implementieren. Die Funktionalität der Implementierung wurde mit generierten sowie realen Aufzeichnungen von Handtrajektorien validiert, und die Robustheit des Algorithmus durch eigene Maßnahmen verbessert. Die Ergebnisse zeigen, dass durch eine optimale Positionierung der Ziele präzise Vorhersagen erzielt werden, was die Anwendung in realen Szenarien unterstützt.

Inhaltsverzeichnis

Inhaltsverzeichnis	3
1 Einleitung	5
1.1 Motivation	6
1.2 Zielsetzung der Arbeit	6
1.3 Aufbau der Arbeit	7
2 Stand der Forschung	8
2.1 Begriffsabgrenzung in der Forschung	8
2.2 Arten der Bewegungsvorhersage	9
2.2.1 Kontextbasierte Ansätze	9
2.2.2 Modellierungsansätze zur Bewegungsvorhersage	10
2.2.3 Vorteile des verfolgten Ansatzes	12
3 Theorie und Implementierung	12
3.1 Theoretische Grundlagen des Algorithmus	13
3.1.1 Modellierung der Trajektorien	13
3.1.2 Auswertung der Modelltrajektorien	14
3.1.3 Berechnung der Wahrscheinlichkeiten	16
3.2 Schwächen des Verfahrens	18
3.3 Ansätze zur Lösung der Schwächen in der Implementierung	19
4 Evaluierung der Implementierung	20
4.1 Testen mit generierten Trajektorien	20
4.1.1 Ergebnisse	23
4.2 Testen mit aufgezeichneten Trajektorien	25
4.2.1 Ergebnisse	26
4.3 Testen mit Trajektorien aus der Studie	28
4.3.1 Aufbau der Studie und Ablauf des Tests	28
4.3.2 Ergebnisse	29
4.3.3 Verbesserungsmöglichkeiten der Vorhersage am Beispiel der Studie	31
4.4 Zusammenfassung der Ergebnisse	32

5 Abschluss	33
5.1 Industrielle Anwendungsmöglichkeiten	33
5.2 Mögliche Weiterentwicklungen	33
5.3 Zusammenfassung	34
Literatur	35
Abbildungsverzeichnis	37

1 Einleitung

Herkömmliche Maschinen, die zur Massenproduktion in der Industrie eingesetzt werden, schränken die Produktionsmöglichkeiten auf repetitive und einfache Aufgaben ein. Um auch komplexere Aufgaben bewältigen zu können, wird das kognitive Verständnis des Menschen benötigt. Die traditionelle Sichtweise, den Arbeitsraum von Mensch und Maschine durch einen Schutzzaun strikt zu trennen, hat sich mit der Einführung kollaborativer Roboter grundlegend verändert. Das Ziel der Mensch-Roboter-Kollaboration ist es, in einem gemeinsamen Arbeitsraum die Präzision, Kraft und Geschwindigkeit des Roboters mit dem Denkvermögen, der Intuition und der Kreativität des Menschen zu kombinieren. Dadurch werden nicht nur die Flexibilität und Effizienz in der Produktion erhöht, sondern auch die Arbeitsbedingungen des Menschen erheblich verbessert.

In der Zukunft wird eine noch engere Zusammenarbeit zwischen Menschen und Robotern erwartet, die durch fortschreitende künstliche Intelligenz, Rechenleistung und Sensorik ermöglicht wird. Um diese Vision zu verwirklichen und das volle Potenzial der Mensch-Roboter-Kollaboration auszuschöpfen, ist eine der Herausforderungen die Verbesserung der kognitiven Fähigkeiten des Roboters. Darunter fällt unter anderem die Vorhersage menschlicher Handlungen, was für einen Menschen intuitiv möglich ist, aber für den Roboter ein Problem darstellt. Daher gibt es in der Forschung viele Ansätze zur Lösung dieses Problems, von denen einer in der vorliegenden Arbeit analysiert und in Python implementiert wird. [Castro21][Peifer24, Kapitel 1.1] [Peifer24, Kapitel 2.1.3]

1.1 Motivation

Um eine reibungslose, sichere und effektive Interaktion zwischen Mensch und Roboter zu gewährleisten, ist die Intentionsvorhersage ein wesentlicher Bestandteil der Mensch-Roboter-Kollaboration. Sie bezeichnet die Fähigkeit eines Roboters, die Handlung eines Menschen zu beobachten und daraus eine zukünftige Absicht oder ein Ziel abzuleiten, bevor die Handlung abgeschlossen ist. Dies ist besonders wichtig, da eine frühzeitige und genaue Vorhersage menschlicher Handlungen es dem Roboter ermöglicht, proaktiv und effizient zu reagieren.

Die Umsetzung von Algorithmen, die menschliche Absichten vorhersagen, stellt erhebliche Herausforderungen bei der Implementierung dar. Denn es muss sichergestellt werden, dass die Verarbeitung von Informationen sowohl im Speicheraufwand als auch in der Rechenzeit effizient ist, damit diese auch in Echtzeit von einem Roboter angewendet werden kann. Die Intentionsvorhersage sollte weiterhin so universell wie möglich sein, um sie in verschiedenen Arbeitsumgebungen und mit unterschiedlichen Personen verwenden zu können, ohne dass umfangreiche Anpassungen erforderlich sind.

1.2 Zielsetzung der Arbeit

Die vorliegende Arbeit befasst sich mit der Herausforderung, die zukünftige Handposition eines Menschen vorherzusagen. Dazu wird ein geeigneter Ansatz in Python implementiert, der zu Beginn einer Handbewegung die wahrscheinlichsten Ziele voraussagt, die von der Hand angestrebt werden. Die Funktionalität wird anhand von Daten aus einer vorhandenen Nutzerstudie sowie eigenen Testfällen evaluiert. Dabei sollen Stärken und Schwächen des ausgewählten Ansatzes analysiert und durch eigene Erweiterungen verbessert werden. Zudem wird die industrielle Anwendung durch diese Tests mit verschiedenen Szenarien berücksichtigt, wobei unterschiedliche Anordnungen der Zielpositionen untersucht werden.

1.3 Aufbau der Arbeit

Als Erstes wird im Kapitel 2 der Stand der Forschung vorgestellt. Darin wird der verfolgte Ansatz im Kontext der Vorhersage menschlicher Handlungen kategorisiert und mit anderen Ansätzen verglichen. In Kapitel 3 wird dann der Ansatz im Detail erläutert und der Umgang mit dessen Schwächen beschrieben. Als Nächstes wird in Kapitel 4 die Implementierung mit verschiedenen Testmethoden evaluiert. Darin werden unter anderem die Ergebnisse mit den Daten aus der Nutzerstudie ausgewertet und wichtige Erkenntnisse festgehalten. Als Letztes werden in Kapitel 5 die Anwendung des Ansatzes im industriellen Kontext sowie weitere Möglichkeiten zur Verbesserung diskutiert und mit einer Zusammenfassung der Arbeit abgeschlossen.

2 Stand der Forschung

2.1 Begriffsabgrenzung in der Forschung

In wissenschaftlichen Arbeiten werden zur Vorhersage menschlicher Handlungen unterschiedliche Begriffe verwendet, die untereinander sehr ähnlich sind und teilweise auch synonym verwendet werden. Zur Abgrenzung werden im Folgenden diese Begriffe genauer definiert und das dazugehörige Problem als Fragestellung formuliert.

- **Activity Recognition** ist die Erkennung einer übergeordneten Handlung, die aus mehreren abgeschlossenen Handlungen besteht.[Subetha16]
Welche übergeordnete Handlung führt der Mensch gerade aus?
- **Action Recognition**¹ ist die Erkennung einer Handlung, nachdem sie vollständig ausgeführt wurde.[Kong22]
Welche Handlung hat der Mensch ausgeführt?
- **Action Prediction**² ist die Vorhersage einer Handlung, bevor sie vollständig ausgeführt wurde.[Kong22]
Welche Handlung wird gerade ausgeführt?
- **Intention Recognition/Prediction**³ ist die Vorhersage einer Absicht oder eines Ziels, basierend auf der Beobachtung aktueller Handlungen, um daraus eine zukünftige Handlung oder Absicht zu schließen.[Kong22][Khan23]
*Welche Absicht oder welches Ziel wird der Mensch verfolgen? bzw.
Welche Handlung wird als Nächstes ausgeführt?*
- **Intent Prediction** ist die Art und Weise, wie der Mensch eine Handlung ausführt, um ein Ziel zu erreichen.[Formica21]
Wie wird das nächste Ziel erreicht? und welches Ziel wird erreicht?

Der verfolgte Ansatz, der am Ende dieses Kapitels eingeführt wird, gehört zum Bereich der **Intent Prediction**.

¹ [Hu16] bezeichnen *Action Recognition* als *online*, wenn die Handlung während der Ausführung erkannt wird, und als *offline*, wenn die Handlung bereits abgeschlossen ist.

² [Kong22] kategorisieren *Action Prediction* in eine kurze Zeitspanne über Sekunden und eine lange Zeitspanne über Minuten, wobei Letzteres auch als *Intention Prediction* bezeichnet wird.

³ Obwohl *Intention Recognition* und *Intention Prediction* unterschiedliche Nuancen haben, wurden sie hier zusammengefasst. Da in beiden Fällen der Fokus auf der Vorhersage der nächsten Handlung liegt.

2.2 Arten der Bewegungsvorhersage

Einen Überblick über die verschiedenen Ansätze zur Vorhersage von menschlichen Handlungen bietet die Arbeit von Rudenko et al. (2020).

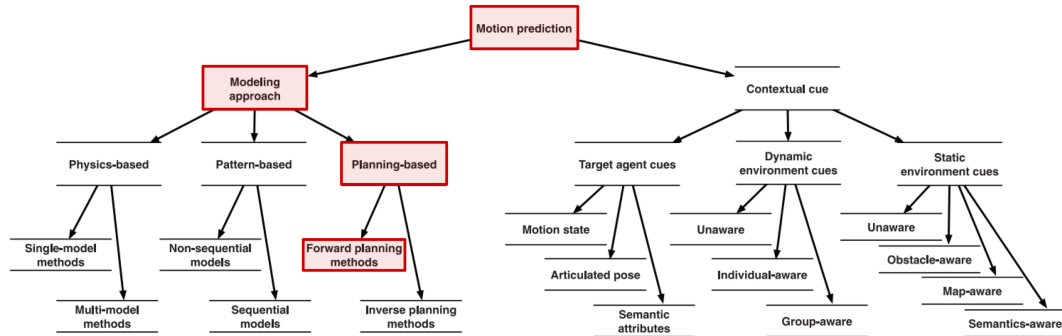


Abb. 1: Überblick über die verschiedenen Ansätze der Bewegungsvorhersage aus [Rudenko20].

Abbildung 1 zeigt, dass sich die Bewegungsvorhersage in zwei Bereiche unterteilt. Auf der rechten Seite sind die kontextuellen Hinweise, die Informationen aus der Umgebung und von den Agenten⁴ nutzen, um Vorhersagen über menschliche Bewegungen zu treffen. Auf der linken Seite sind die Modellansätze, die ein Modell entwerfen, das eine menschliche Bewegung beschreibt, um daraus Vorhersagen abzuleiten. Darin lässt sich auch der in dieser Bachelorarbeit verfolgte Ansatz zuordnen, der in dieser Abbildung in rot hervorgehoben ist. [Rudenko20]

2.2.1 Kontextbasierte Ansätze

Die Kategorie der kontextbasierten Ansätze wird in dieser Arbeit nicht verwendet, weswegen nur kurz auf deren Bedeutung eingegangen wird. Diese betrachten alle möglichen internen und externen Faktoren, die die Bewegungen des Zielagenten⁵ beeinflussen können. Interne Faktoren können beispielsweise die Kopfhaltung oder das Augentracking sein, um alle Ziele im Blickfeld des Agenten zu betrachten und damit die Wahrscheinlichkeit einer korrekten Vorhersage zu erhöhen. Externe Faktoren können Veränderungen in der Umgebung sein, die weitere Hinweise auf das nächste Ziel geben. [Rudenko20]

⁴ Mit Agenten sind hier dynamische Objekte wie Menschen oder Fahrzeuge gemeint.

⁵ Agent, dessen Bewegung vorhergesagt werden soll.

2.2.2 Modellierungsansätze zur Bewegungsvorhersage

Rudenko et al. teilen den Modellierungsansatz in drei Kategorien: physikbasiert, musterbasiert und planbasiert. Für die Vorhersage von Handbewegungen werden hauptsächlich muster- und planbasierte Ansätze angewendet, während physikbasierte Modelle⁶ in der Regel nicht verwendet werden.[Rudenko20]

Musterbasierte Ansätze nutzen Trainingsdaten von Handtrajektorien, um mithilfe von maschinellern Bewegungsmustern zu identifizieren. Die Vorhersage von Handbewegungen basiert in diesen Methoden auf den zuvor beobachteten und erlernten Handtrajektorien. Ein Beispiel für eine sequentielle Version der musterbasierten Methode ist der Ansatz von Claudia Pérez-D'Arpino und Julie A. Shah. Als Erstes werden menschliche Greifbewegungen durch 20 Demonstrationen für jede Bewegungsklasse aufgenommen. Eine Bewegungsklasse ist dabei die Trajektorie der Hand, die von einer Startposition zu einer Zielposition verläuft, und die Demonstrationen dazu beinhalten die Position des Handgelenks sowie die Rotationen der Gelenke, die anschließend in einer Bibliothek abgespeichert werden. Diese Daten bilden dann die Grundlage, um während der Offline-Lernphase Bewegungsmuster zu erkennen und zu speichern. Aus dieser Bibliothek werden schließlich mehrere statistische Modelle erstellt, die wiederum zur Vorhersage von Zielen in Echtzeit genutzt werden. In der praktischen Anwendung wird die aktuell beobachtete Bewegung mit allen relevanten Modellen in der Bibliothek verglichen und daraus das wahrscheinlichste Ziel abgeleitet.

[Rudenko20][Pérez-D'Arpino15]

Planbasierte Ansätze hingegen verwenden einen Plan im Rahmen mathematischer Gleichungen. Dazu werden direkte Annahmen über den zukünftigen Verlauf einer Handbewegung getroffen, die in Form einer vorausgesagten Trajektorie modelliert wird. Bei der Rückwärtsplanung muss diese mathematische Gleichung in Form einer Kostenfunktion zuerst geschätzt werden, wobei als Beispiel der Ansatz aus der Arbeit von Jim Mainprice, Rafi Hayne und Dmitry Berenson betrachtet wird. Dieser Ansatz besteht aus folgenden zwei Schritten: Im ersten Schritt werden sieben grobe Bewegungen⁷ über die Zeit während der Bearbeitung einer gemeinsamen Aufgabe mit einer zweiten Person

⁶ Physikbasierte Modelle beschreiben die zukünftige Bewegung eines Menschen auf Grundlage fester physikalischer Gesetze. Beispielsweise werden hier Modelle erstellt unter der Annahme von konstanter Geschwindigkeit oder konstanter Beschleunigung.

⁷ Mit „groben Bewegungen“ sind mehrere Bewegungen bzw. Trajektorien in einer längeren Aufzeichnung gemeint.

aufgezeichnet, die anschließend manuell in einzelne elementare Bewegungen⁸ segmentiert werden. Mit der Annahme, dass diese segmentierten Bewegungen optimal sind, wird dann unter Verwendung des *Inverse Optimal Control*-Algorithmus eine Kostenfunktion erlernt, die diese gespeicherten Trajektorien am besten beschreibt. Die Eingaben dieser Kostenfunktion sind zwei Merkmale mit jeweiligen Gewichten. Das eine Merkmal beschreibt die Abstände zwischen den Gelenken des Menschen, um Kollisionen zu vermeiden, und das andere Merkmal ist die Glätte der Trajektorie, um eine natürliche Bewegung zu modellieren. Im zweiten Schritt wird diese Kostenfunktion benutzt und auf ein kinematisches Modell eines Menschen angewendet, um daraus wiederum eine Handtrajektorie vorherzusagen. Bei der Anwendung werden in Echtzeit diese vorhergesagten Trajektorien iterativ neu geplant und an die aktuelle Bewegung des Menschen angepasst.[Rudenko20][Mainprice15]

Im Gegensatz zur Rückwärtsplanung sind in der Vorwärtsplanung die Parameter der Gleichung zur Modellierung von vorhergesagten Handtrajektorien bereits festgelegt und müssen nicht davor approximiert werden. Der in dieser Arbeit verfolgte Ansatz von Zanchettin und Rocco fällt in diese Kategorie. Das Paper stellt ein neues modellbasiertes Verfahren zur Vorhersage von bekannten Zielen anhand einer initialen Handbewegung eines Menschen vor, welches auf der Bayesschen Statistik basiert. Dies wird im nächsten Kapitel detailliert erläutert, weswegen es im Folgenden nur kurz beschrieben wird. Als Erstes wird anhand von zwei zeitlich aufeinanderfolgenden Messungen der Handgelenkposition für jedes Ziel eine vorhergesagte Trajektorie erstellt, welche die zukünftige Bewegung der Hand darstellt. Im nächsten Schritt werden diese Trajektorien mit der zuletzt gemessenen Handposition ausgewertet. Diese Auswertung stellt eine Stichprobe dar, die über mehrere Messungen gesammelt wird und im letzten Schritt die Wahrscheinlichkeit für die einzelnen Ziele bildet.[Rudenko20][Zanchettin17]

⁸ Mit „elementaren Bewegungen“ sind genaue, einzelne Bewegungen gemeint, wie die Bewegung einer Hand von der Ruheposition zum Greifen eines Objekts. Bei der Validierung sind dabei 33 dieser elementaren Bewegungen entstanden.

2.2.3 Vorteile des verfolgten Ansatzes

Die Auswahl dieses Ansatzes zur Vorhersage basiert auf folgenden Vorteilen gegenüber den zuvor beschriebenen Methoden:

- **Universalität:** Da der Ansatz lediglich auf den neuesten Messungen basiert, müssen zuvor keine aufwendigen Trainingsdaten aufgenommen, analysiert und trainiert werden. Dies hat zur Folge, dass der Erfolg einer korrekten Vorhersage nicht von der Qualität und Quantität sowie von der Person und der Umgebung der zuvor gesammelten Daten abhängt, sondern flexibel in verschiedenen Szenarien angewendet werden kann.
- **Reaktionsfähigkeit:** Die Betrachtung der neuesten Messungen hat zusätzlich den Vorteil, dass Umentscheidungen zu einem neuen Ziel schnell erkannt werden können.
- **Effizienz:** Es müssen keine Optimierungen während der Laufzeit berechnet und keine weiteren Daten abgespeichert werden, was die Anwendbarkeit in Echtzeit erheblich vereinfacht.
[Zanchettin17]

Trotz dieser Vorteile weist der Ansatz auch Schwächen auf, die genauer in Kapitel 3.2 erläutert werden.

3 Theorie und Implementierung

In diesem Kapitel wird zunächst der theoretische Teil des Ansatzes von Zanchettin und Rocco (2017) im Detail erläutert. Da im Kapitel davor bereits die Stärken gegenüber anderen Verfahren behandelt wurden, wird danach nur auf die Schwächen eingegangen und anschließend der Umgang mit diesen Schwächen bei der Implementierung erläutert.[Zanchettin17]

In meiner Arbeit wurde dieser theoretische Ansatz in eine praktische Implementierung in Python überführt und mit aufgezeichneten Handtrajektorien in Echtzeit simuliert. Die Funktionalität wurde anhand von mehreren Tests validiert, die im Kapitel 4 genauer vorgestellt werden.

3.1 Theoretische Grundlagen des Algorithmus

Zu Beginn sind alle betrachteten Ziele \mathcal{G} mit $p_G \in \mathcal{G}$ gegeben und es werden kontinuierlich dreidimensionale Koordinaten des Handgelenks p_n zum Zeitpunkt t_n gemessen. Die Bewegungsrichtung der aktuellen Messung kann als normierter Richtungsvektor p'_n wie folgt berechnet werden:

$$p'_n = \frac{p_n - p_{n-1}}{\|p_n - p_{n-1}\|} \quad (1)$$

3.1.1 Modellierung der Trajektorien

Um die zukünftige Bewegung der Hand zu jedem Ziel vorherzusagen, wird für jedes Ziel eine Trajektorie modelliert. Diese Trajektorien beschreiben, wie sich die Hand voraussichtlich zu den jeweiligen Zielen bewegen wird, und werden durch ein Polynom dritten Grades als Kurve im dreidimensionalen Raum dargestellt. Diese Kurve ist wie folgt definiert:

$$\hat{p}(s_n | p_G, x_{n-1}) = \sum_{k=0}^3 a_k \cdot s_n^k \quad (2)$$

mit den Koeffizienten:

$$\begin{aligned} a_0 &= p_{n-1} \\ a_1 &= p'_{n-1} \\ a_2 &= 1.5p_G - 1.5p_{n-1} - 1.5p'_{n-1} \\ a_3 &= -0.5p_G + 0.5p_{n-1} + 0.5p'_{n-1} \end{aligned}$$

Diese Koeffizienten wurden so bestimmt, dass die Bedingungen $0 \leq s_n \leq 1$, mit $s_n = 0$ für p_{n-1} und $s_n = 1$ für p_G , erfüllt sind. Das bedeutet, dass die Kurve von der zuletzt gemessenen Handposition p_{n-1} bis zum betrachteten Ziel p_G verläuft und somit die vorhergesagte Trajektorie beschreibt. Darüber hinaus wurde durch die Minimierung der Bogenlänge erreicht, dass die Kurve glatt verläuft, um eine natürliche Handbewegung zum Ziel zu modellieren. Die Funktion $\hat{p}(s_n | p_G, x_{n-1})$ stellt somit die vorhergesagte Handposition entlang des Pfads zum betrachteten Ziel p_G dar, basierend auf der letzten Messung x_{n-1} und dem Fortschritt s_n .

3.1.2 Auswertung der Modelltrajektorien

Im nächsten Schritt werden die vorhergesagten Trajektorien mit der aktuellen Handposition ausgewertet, indem die Winkel zwischen gemessenen und vorhergesagten Richtungsvektoren berechnet werden. Sobald eine neue Messung p_n vorliegt, kann der Fortschritt s_n auf dem Pfad zum betrachteten Ziel folgendermaßen approximiert werden:

$$s_n = \frac{\|p_n - p_{n-1}\|}{\|p_n - p_{n-1}\| + \|p_G - p_n\|} \quad (3)$$

Diese Approximation stellt sicher, dass s_n im Intervall $[0, 1]$ liegt und der Wert relativ zum Abstand von der vorherigen Handposition und dem betrachteten Ziel berechnet wird. Das bedeutet, je weiter die aktuelle Messung p_n von der vorherigen p_{n-1} in Richtung des Ziels p_G fortgeschritten ist, desto näher liegt s_n an der oberen Intervallgrenze. Umgekehrt: Je geringer der Fortschritt, desto näher liegt s_n an der unteren Intervallgrenze.

Um die vorhergesagte Bewegungsrichtung zu berechnen, wird das Polynom aus der Gleichung 2 nach der Variable s_n abgeleitet und das approximierte s_n eingesetzt. Auf diese Weise ergibt sich der vorhergesagte Richtungsvektor $\hat{p}'(s_n|p_G, x_{n-1})$.

Schließlich werden die Winkel $\theta_i \in \Theta$ zwischen $\hat{p}'(s_n|p_G, x_{n-1})$ und p'_n aus Gleichung 1 auf der XY-Ebene berechnet, wobei jeder Winkel θ_i dem entsprechenden Ziel p_{G_i} zugeordnet ist.

Anmerkung: Die expliziten Berechnungen von p'_n aus Gleichungen 1 und s_n aus Gleichung 3 wurden von mir ergänzt und sind so nicht direkt im Paper gegeben.

Anschaulich lassen sich die Berechnungen bis hierhin mit dem folgenden Beispiel visualisieren:

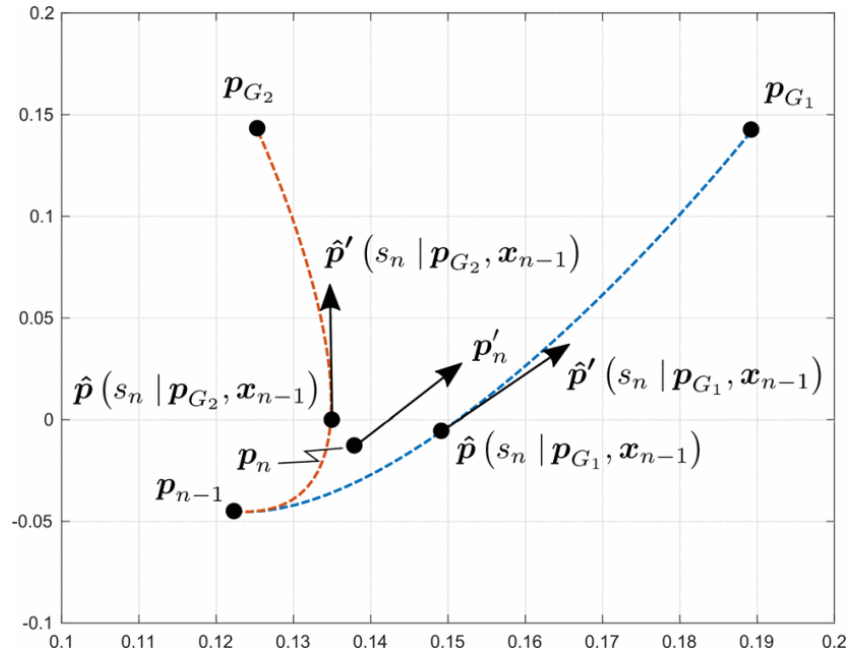


Abb. 2: Beispiel der Modellierung von zwei vorhergesagten Trajektorien zu den jeweiligen Zielen in der XY-Ebene aus [Zanchettin17].

- **Gegebene Daten:**

- p_{G_1} und p_{G_2} sind die Koordinaten der Ziele.
- p_n und p_{n-1} sind die gemessenen Handgelenkpunkte, jeweils zum Zeitpunkt t und $t - 1$, sowie der aus Gleichung 1 resultierende Richtungsvektor p'_n .

- **Daten aus der modellierten Trajektorie:**

- $\hat{p}(s_n | p_{G_1}, x_{n-1})$ in blau und $\hat{p}(s_n | p_{G_2}, x_{n-1})$ in rot sind die vorhergesagten Modelltrajektorien aus Gleichung 2.
- Der schwarze Punkt auf den Trajektorien stellt den jeweiligen Fortschritt s_n aus Gleichung 3 dar.
- $\hat{p}'(s_n | p_{G_1}, x_{n-1})$ und $\hat{p}'(s_n | p_{G_2}, x_{n-1})$ sind die entsprechenden vorhergesagten Richtungsvektoren am Punkt s_n .

3.1.3 Berechnung der Wahrscheinlichkeiten

Im letzten Schritt werden die Wahrscheinlichkeiten für jedes Ziel ausgewertet. Dazu wird aus den Winkeln Θ die Varianz σ_Θ^2 berechnet und mit dem Mittelwert $\mu = 0$ eine Normalverteilung $\mathcal{N}(0, \sigma_\Theta^2)$ aufgestellt. Dann wird mit der Wahrscheinlichkeitsdichtefunktion⁹ $f_{\mathcal{N}}(\theta_i; 0, \sigma_\Theta)$ die Wahrscheinlichkeit für jeden Winkel θ_i berechnet. Ein Beispiel für $f_{\mathcal{N}}$ mit unterschiedlichen Varianzen ist in der folgenden Abbildung dargestellt:

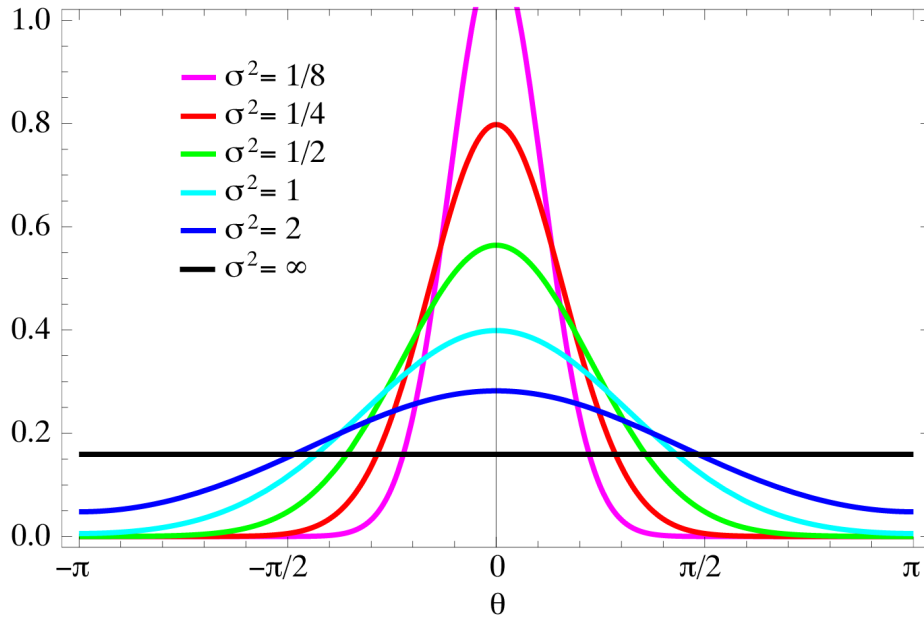


Abb. 3: Wahrscheinlichkeitsdichtefunktion mit Mittelwert $\mu = 0$ und mehreren Varianzen σ^2 im Vergleich aus [PAR11].

In diesem Kontext bedeutet ein kleinerer Winkel θ_i , dass die Wahrscheinlichkeit für diesen Winkel höher ist. Also je ähnlicher der gemessene Richtungsvektor p'_n und der vorhergesagte Richtungsvektor $\hat{p}'(s_n|p_G, x_{n-1})$ sind, desto wahrscheinlicher ist das entsprechende Ziel zu diesem Zeitpunkt bzw. zu dieser Stichprobe. Da zudem die Varianz der Winkel berücksichtigt wird, kann die Wahrscheinlichkeit relativ zur Streuung der Winkel bewertet werden. Das bedeutet, dass eine niedrige Varianz zu hohen Wahrscheinlichkeiten in der Nähe vom Mittelwert $\mu = 0$ führt, während eine hohe Varianz zu geringen Wahrscheinlichkeiten führt.

⁹ Verwendete Wahrscheinlichkeitsdichtefunktion: $f_{\mathcal{N}}(\theta_i; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(\theta_i - \mu)^2}{2\sigma^2}\right)$

Damit lässt sich folgende iterative Funktion zur Berechnung der Wahrscheinlichkeit für jedes Ziel definieren:

$$\tilde{\mathcal{P}}_n(p_G) = \mathcal{P}_{n-1}(p_G) \cdot f_{\mathcal{N},n}(\theta_i; 0, \sigma_\Theta) \quad (4)$$

Dabei gilt:

- $\tilde{\mathcal{P}}_n(p_G)$ ist die unnormierte Wahrscheinlichkeit des Ziels p_G zum Zeitpunkt t_n .
- $\mathcal{P}_{n-1}(p_G)$ ist die normierte Wahrscheinlichkeit des Ziels p_G zum Zeitpunkt t_{n-1} .
- $f_{\mathcal{N},n}(\theta_i; 0, \sigma_\Theta)$ ist die Wahrscheinlichkeit des Winkels θ_i für das Ziel p_G zum Zeitpunkt t_n .

Die Normierung wird folgendermaßen berechnet:

$$\forall p_G \in \mathcal{G} : \mathcal{P}_n(p_G) = \frac{\tilde{\mathcal{P}}_n(p_G)}{\max\left(1, \sum_{p_G \in \mathcal{G}} \tilde{\mathcal{P}}_n(p_G)\right)} \quad (5)$$

Wenn die Summe aller normierten Wahrscheinlichkeiten kleiner 1 ist, wird die Wahrscheinlichkeit, dass keines der Ziele angestrebt wird, als nicht kategorisiertes Ziel $\neg\mathcal{P}_n$ berechnet:

$$\neg\mathcal{P}_n = 1 - \sum_{p_G \in \mathcal{G}} \mathcal{P}_n(p_G) \quad (6)$$

Bei Betrachtung von Abbildung 2 zeigt sich, dass die Wahrscheinlichkeit von p_{G_1} zum Zeitpunkt t_n höher ist als die von p_{G_2} , da der Winkel zwischen $\hat{p}(s_n | p_{G_1}, x_{n-1})$ und p'_n kleiner ist als der zwischen $\hat{p}(s_n | p_{G_2}, x_{n-1})$ und p'_n . Unter der Annahme, dass die zukünftigen Bewegungsrichtungen p' gleich bleiben werden, lässt sich nach Gleichung 4 schließen, dass für folgende Stichproben die Wahrscheinlichkeit von p_{G_1} weiter steigen und die von p_{G_2} weiter sinken wird.

3.2 Schwächen des Verfahrens

Trotz der zahlreichen Stärken des Verfahrens, die in Kapitel 2.2.2 aufgezählt wurden, gibt es auch signifikante Schwächen. Aufgrund der Modellierung mit nur zwei Messpunkten besteht eine der größten Herausforderungen darin, den Algorithmus gegenüber Rauschen robuster zu machen.

Das zweite Problem besteht in der Modellierung der Trajektorien.

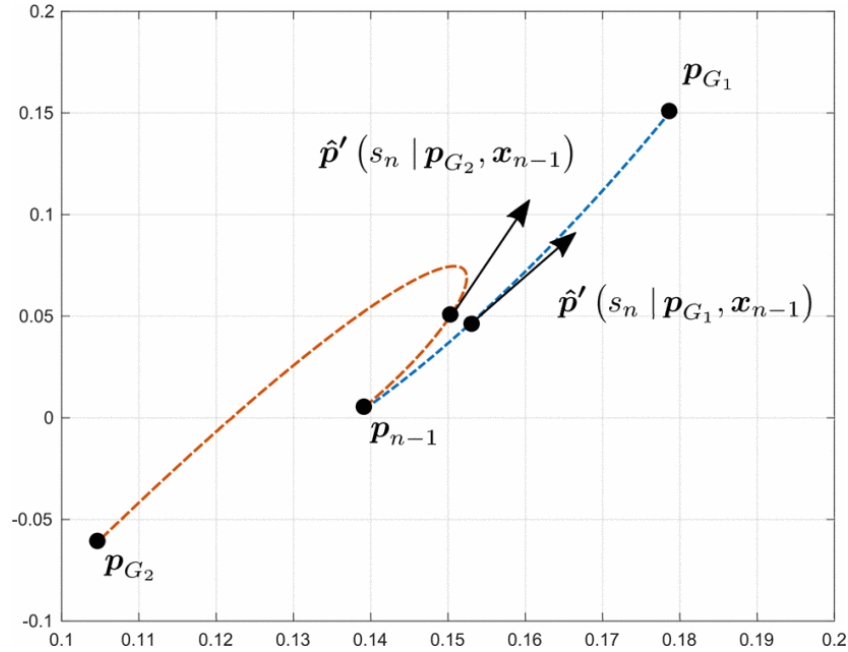


Abb. 4: Beispiel aus [Zanchettin17], in der zwei Ziele durch ähnliche Richtungsvektoren fälschlicherweise die gleiche Wahrscheinlichkeit haben.

Wenn die Messung p_{n-1} zwischen zwei Zielen liegt, kann es vorkommen, dass durch die Ähnlichkeit der vorhergesagten Richtungsvektoren beide Ziele die gleiche Wahrscheinlichkeit aufweisen, obwohl sich die Hand klar zu einem dieser Ziele bewegt. Dieses Problem wird zunehmend größer mit steigender Anzahl der Ziele, was zum letzten Problem führt.[Zanchettin17]

Die Korrektheit der Vorhersage reduziert sich mit der Anzahl der betrachteten Ziele, da der Abstand zwischen diesen geringer wird. Dies führt zu sehr ähnlichen Trajektorien und somit zu Richtungsvektoren, die in die gleiche Richtung zeigen. Daraus folgt schließlich, dass mehrere Ziele gleich geringe Wahrscheinlichkeiten haben.

3.3 Ansätze zur Lösung der Schwächen in der Implementierung

Um mit den beschriebenen Schwächen umzugehen, wurden mehrere Maßnahmen ergriffen. Das Problem des Rauschens wurde durch die Implementierung von drei Methoden zur Glättung der Daten angegangen: *Simple Moving Average* (SMA), *Weighted Moving Average* (WMA) und *Exponential Moving Average* (EMA). Diese Methoden wurden in das Programm integriert, um ungenaue Messungen in Echtzeit zu glätten¹⁰. Im Folgenden werden die Strategien kurz erläutert:

- **SMA** berechnet den Durchschnitt einer festen Anzahl von Datenpunkten. Diese Methode ist besser geeignet für längere Trajektorien und reagiert nicht schnell auf Veränderungen.
- **WMA** ist SMA aber mit Gewichten für jeden Datenpunkt, damit relevante Datenpunkte stärker berücksichtigt werden können. In der Implementierung wurden jüngere Datenpunkte stärker gewichtet als ältere, wodurch diese Methode schneller auf Änderungen reagieren kann als der SMA.
- **EMA** verwendet ebenfalls Gewichte, die jedoch exponentiell mit zunehmendem Alter der Daten abnehmen. Diese Methode ermöglicht die schnellste Reaktion auf Änderungen in den Daten.[Lotysh23]

Für das Problem mit fälschlicherweise ähnlichen Richtungsvektoren wurde ein minimaler Fortschritt s_n festgelegt, da Richtungsvektoren, die zu nah am Beginn des vorhergesagten Pfads liegen, immer in die ähnliche Richtung zeigen. Des Weiteren werden mit

$$(p_G - p_n)^T \cdot p'_n > 0 \quad (7)$$

ausschließlich Ziele berücksichtigt, die in der Bewegungsrichtung p'_n liegen¹¹.

Zur Verbesserung des letzten Problems, der ungenauen Wahrscheinlichkeit bei zunehmender Anzahl von Zielen, wurde die folgende Kostenfunktion hinzugefügt:

$$K_d(\omega, d) = \frac{1}{1 + \omega \cdot d} \quad \text{mit } \omega, d \geq 0 \quad (8)$$

Dabei sind ω das Gewicht und d die Distanz vom Handgelenk zum betrachteten Ziel. Diese Funktion wird vor der Normierung angewendet und reduziert die Wahrscheinlichkeit von Zielen, die einen größeren Abstand zur Hand haben.

¹⁰ Alternativ könnten die Messdaten auch außerhalb des Programms vorverarbeitet werden. Wichtig ist lediglich, dass eine Glättungsstrategie angewendet wird, da sonst keine verlässlichen Vorhersagen erzielt werden können.

¹¹ Diese Maßnahme wurde im Paper [Zanchettin17] zu diesem Problem als Lösung vorgestellt.

4 Evaluierung der Implementierung

Um die Funktionalität des implementierten Ansatzes zu testen und optimale Werte für die Parameter sowie eine geeignete Konfiguration an Zielpositionen zu finden, wurden drei verschiedene Testmethoden aufgestellt und dessen Ergebnisse ausgewertet. Dieses Kapitel ist folgendermaßen aufgebaut:

In Kapitel 4.1 liegt der Fokus auf der Bestätigung der Funktionalität der Implementierung mit verschiedenen Zielkonfigurationen und das Finden optimaler Parameter mithilfe generierter Trajektorien. Kapitel 4.2 konzentriert sich darauf, ideale Zielkonfigurationen zu finden und den Algorithmus mit aufgezeichneten Handpositionen für die praktische Anwendung zu validieren. Im letzten Test in Kapitel 4.3 werden mithilfe von Daten aus der Studie die Grenzen der Implementierung ermittelt. Dabei liegt der Fokus auf einer hohen Anzahl an Zielen die mit Aufzeichnungen von verschiedenen Personen getestet wird.

4.1 Testen mit generierten Trajektorien

Der erste Test mit generierten Trajektorien wurde verwendet, um kontinuierlich während der Implementierung die Funktionalität zu prüfen und am Ende optimale Parameter zu finden. Für die Auswahl der Positionen der Ziele gibt es viele Möglichkeiten und noch mehr Optionen, um Trajektorien zu diesen Zielen zu modellieren.

Abbildung 5 zeigt alle Konfigurationen, die getestet wurden. Ein Kasten repräsentiert eine Konfiguration, und ein Punkt markiert die Position eines Ziels. Die Anzahl der Ziele steigt schrittweise, beginnend mit nur einem Ziel bis hin zu sieben Zielen. Dabei ist der Fall mit nur einem Ziel nicht trivial, da auch die Möglichkeit eines nicht kategorisierten Ziels besteht.

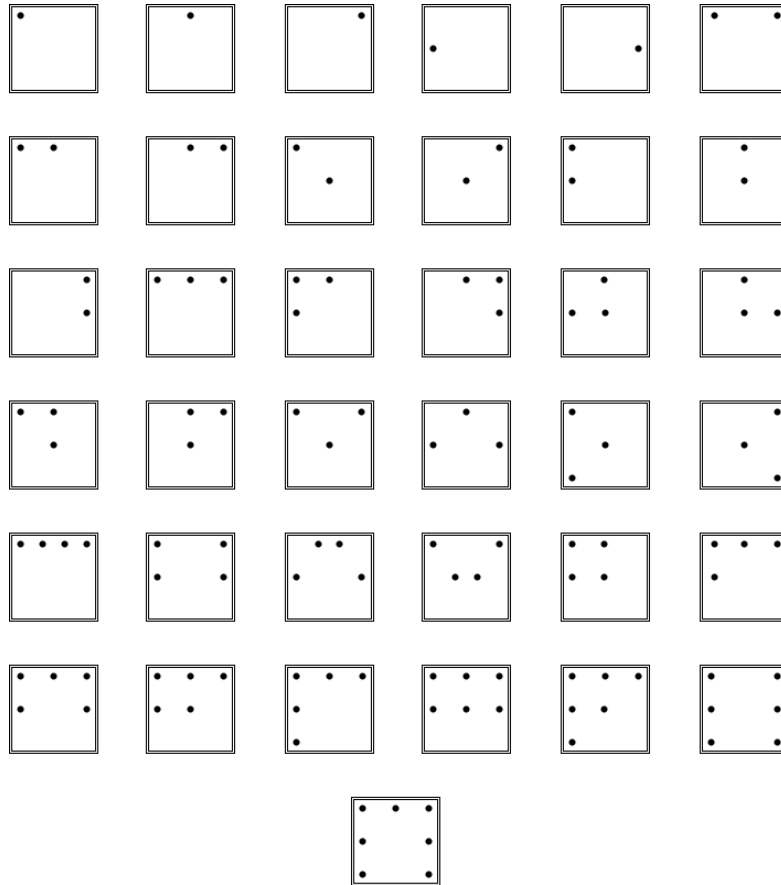


Abb. 5: Zielkonfigurationen für die Trajektorien generiert wurden.

Als Nächstes wurden die Handbewegungen in jeder Konfiguration zu jedem Ziel mehrmals simuliert. Diese Trajektorien wurden mittels quadratischen Bézierkurven mit drei Kontrollpunkten erstellt, wobei diese Punkte jeweils am Start- und Endpunkt sowie in der Mitte der Linie positioniert sind.

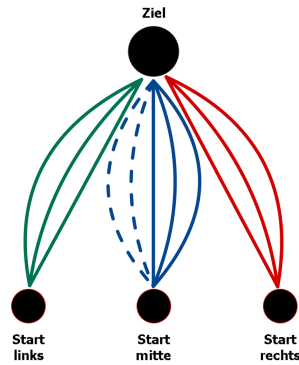


Abb. 6: Skizze generierte Trajektorien zu einem Ziel.

Abbildung 6 zeigt als Skizze, wie Trajektorien zu einem Ziel verlaufen. Es gibt immer drei Startpunkte von links nach rechts, und bei jedem dieser drei Startpunkte werden drei Kurven gezeichnet, die unterschiedlich gekrümmt sind. Die gestrichelten Linien in blau bedeuten dabei, dass zufällig die Richtung der Krümmung gewählt wird im Fall, dass Start und Endpunkt sich auf der gleichen X-Achse befinden.

Insgesamt sind 1032 einzelne Tests entstanden, die mit verschiedenen Parametern aus Kapitel 3.3 beschriebenen Maßnahmen inkrementell ausgeführt wurden. Da die generierten Trajektorien durch den Bézier-Algorithmus glatt verlaufen, wurde zusätzlich weißes Rauschen¹² hinzugefügt, um auch ungenaue Messungen in Betracht zu ziehen.

Für die Bewertung des Tests wurden folgende zwei Bedingungen formuliert:

1. Entlang der Trajektorie wurde für das betrachtete Ziel nach mindestens 10 Stichproben eine Wahrscheinlichkeit von 60% erreicht.
2. Entlang der Trajektorie hatte das betrachtete Ziel über alle Stichproben insgesamt eine höhere Wahrscheinlichkeit im Gegensatz zu allen anderen Zielen.

Und das Ziel wurde:

- **Erkannt:** Wenn beide Bedingungen erfüllt wurden.
- **Teilweise erkannt:** Wenn entweder Bedingung 1 oder 2 erfüllt wurde.
- **Nicht erkannt:** Wenn keines der beiden Bedingungen erfüllt wurde.

¹² Nach [Hübner20] hat die Hololens einen mittleren absoluten Trajektorienfehler von 1.9 ± 0.4 cm. Dies wurde mittels Gauß-Verteilung mit Mittelwert gleich 0 und einer Standardabweichung von 0.4 cm simuliert.

4.1.1 Ergebnisse

Die wichtigste Erkenntnis aus den Testergebnissen war die starke Anfälligkeit des Algorithmus gegenüber Rauschen. Ohne eine Strategie zum Glätten der Daten wurde bei keinem der Testfälle ein Ziel erkannt. Im Vergleich konnten durch das Einsetzen einer Strategie etwa 800 Ziele korrekt vorhergesagt werden. Alle Strategien erzielten in diesen Testfällen ähnliche Ergebnisse, wobei WMA als die geringfügig bessere Strategie abschnitt.

Eine weitere Maßnahme, die zu verbesserten Ergebnissen geführt hat, war das Hinzufügen einer unteren Grenze für die Varianz, deren Berechnung in Kapitel 3.1.3 eingeführt wurde.

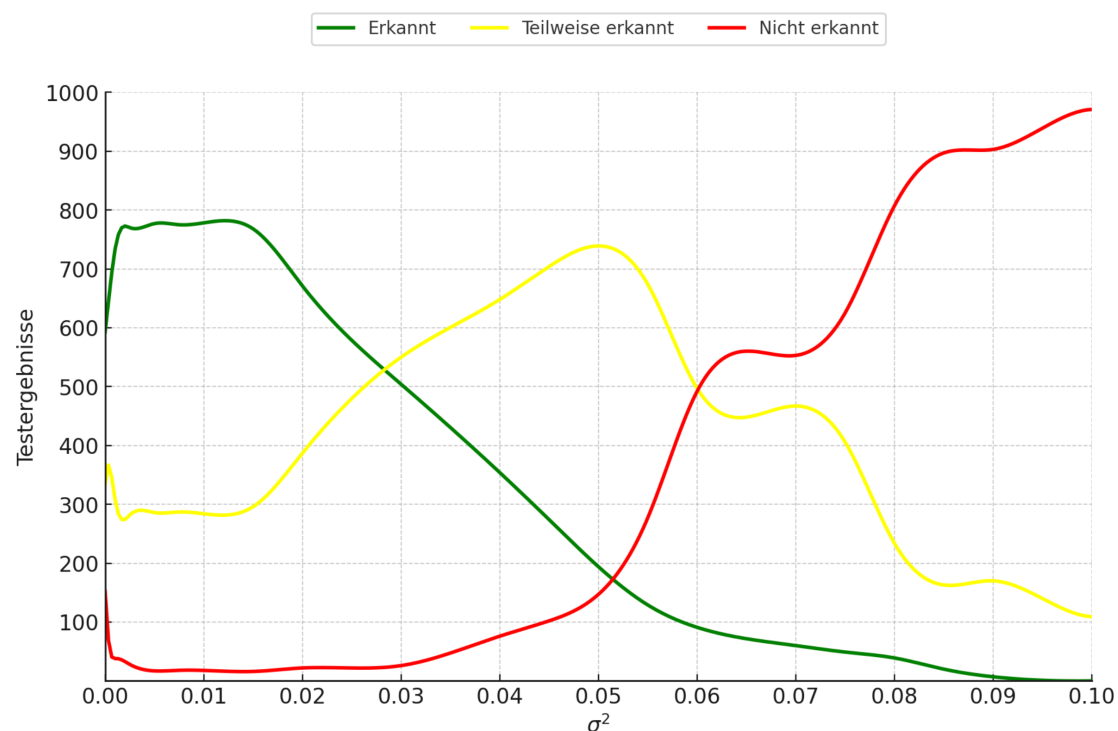


Abb. 7: Erfolgreiche Zielerkennung in Abhängigkeit von der unteren Grenze der Varianz σ^2 .

Durch das Hinzufügen dieser Grenze konnte im Optimum (von 0.005 bis 0.015) eine Verbesserung der Anzahl erkannter Ziele von 31% gemessen werden. Ab einer Varianz von 0.015 sinkt diese Anzahl jedoch stetig ab und ab einer Varianz von 0.03 steigt die Anzahl nicht erkannter Ziele deutlich an.

Als letzte Beobachtung werden Tests betrachtet, in denen das Ziel meistens nicht erkannt wurde. Als Beispiel dazu soll folgende Abbildung dienen:

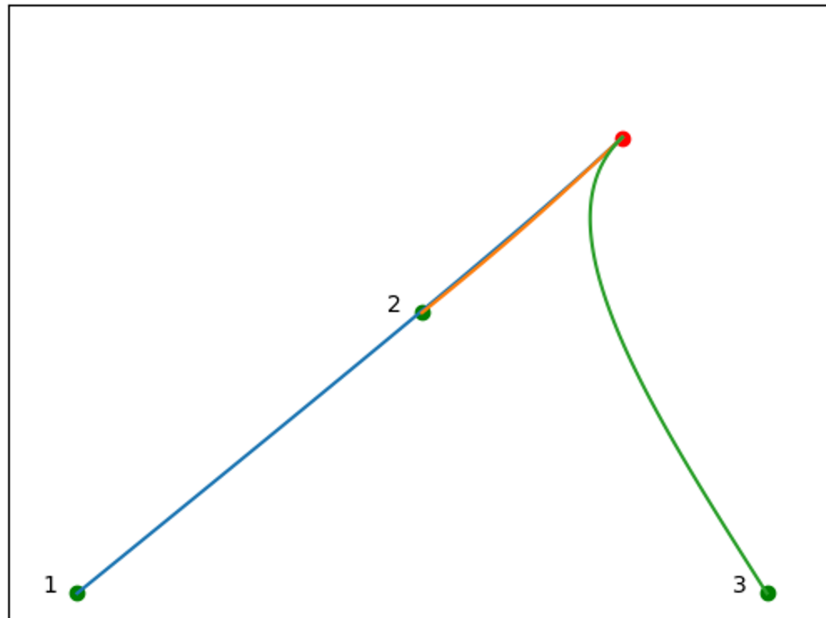


Abb. 8: Fall, in dem sich zwei vorhergesagte Trajektorien zu Ziel 1 und Ziel 2 überdecken.

Durch eine ungünstige Positionierung der Ziele oder bestimmte Bewegung der Hand kann es dazu kommen, dass die vorhergesagten Trajektorien von zwei oder mehreren Zielen sich überdecken. In Abbildung 8 ist dies bei **Ziel 1** in blau und **Ziel 2** in orange zu sehen. Dies hat zur Folge, dass die vorhergesagte Bewegungsrichtung an den überdeckten Stellen nahezu identisch ist, was schließlich dazu führt, dass die Wahrscheinlichkeiten für beide Ziele gleich werden. Die Normierung durch Gleichung 5 sorgt dafür, dass keines der beiden Ziele die Schwelle von 60% überschreitet.

4.2 Testen mit aufgezeichneten Trajektorien

Im zweiten Test geht es um die Validierung der praktischen Anwendbarkeit des Algorithmus und das Finden von geeigneten Zielkonfigurationen. Diese wurden so ausgewählt, dass durch eine gute Positionierung eine hohe Anzahl von Zielen betrachtet werden kann. Zu diesem Zweck wurden zwei Zielkonfigurationen ausgewählt, und für jedes dieser Ziele wurden Handtrajektorien durch die Verwendung von einer Hololens mit einer Abtastrate von 60 Hz aufgezeichnet. Zwischen einzelnen Messungen wurden nur Vorhersagen modelliert, die eine Distanz von mindestens 1 cm zueinander haben.

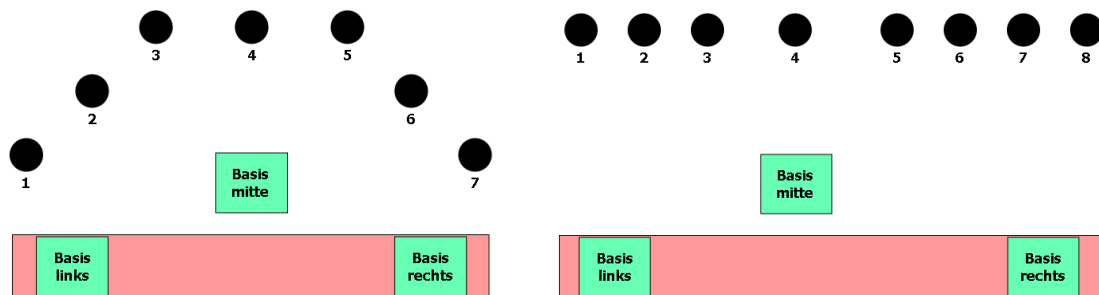


Abb. 9: Links: Skizze der 1. Konfiguration; Rechts: Skizze der 2. Konfiguration

Abbildung 9 zeigt beide Zielkonfigurationen, mit denen getestet wurde. Die schwarzen Punkte sind die einzelnen Ziele und die grünen Kästen sind Startpunkte, von denen die Hand sich zu den einzelnen Zielen bewegt hat. Der rote Bereich beschreibt das nicht kategorisierte Ziel, also den Fall, dass keines der nummerierten Ziele angestrebt wird.

Die meisten Trajektorien starten in „Basis mitte“, welche als kollaborative Arbeitszone¹³ dient, und verlaufen zu jeweils einzelnen Zielen. Neben der direkten Handbewegung zu einem Ziel wurden auch Fälle betrachtet, in denen die begonnene Bewegung kurzfristig zu einem anderen Ziel umgelenkt wurde, was einen der Vorteile dieses Ansatzes darstellt.

Die Kriterien zur Bewertung dieser Testfälle sind ähnlich den generierten Testverfahren. Ein Ziel gilt als *erkannt*, wenn über mehrere¹⁴ Stichproben eine Wahrscheinlichkeit von 60% erreicht wurde, und als *teilweise erkannt*, wenn es zu den drei Zielen mit der höchsten Wahrscheinlichkeit gehört.

¹³ Der Bereich, in dem Mensch und Maschine gemeinsam an einer Aufgabe arbeiten.

¹⁴ Da die Geschwindigkeit der Hand bei aufgezeichneten Handbewegungen variiert, wurden 5 bis 10 Stichproben betrachtet.

4.2.1 Ergebnisse

In den Testergebnissen konnte Konfiguration 1 als gute Positionierung der Ziele bestätigt werden, da 83,1% der Ziele erkannt und nur 4,6% nicht erkannt wurden. In Gegensatz dazu wurden bei Konfiguration 2 lediglich 59,7% der Ziele erkannt und 12,9% der Ziele nicht erkannt.

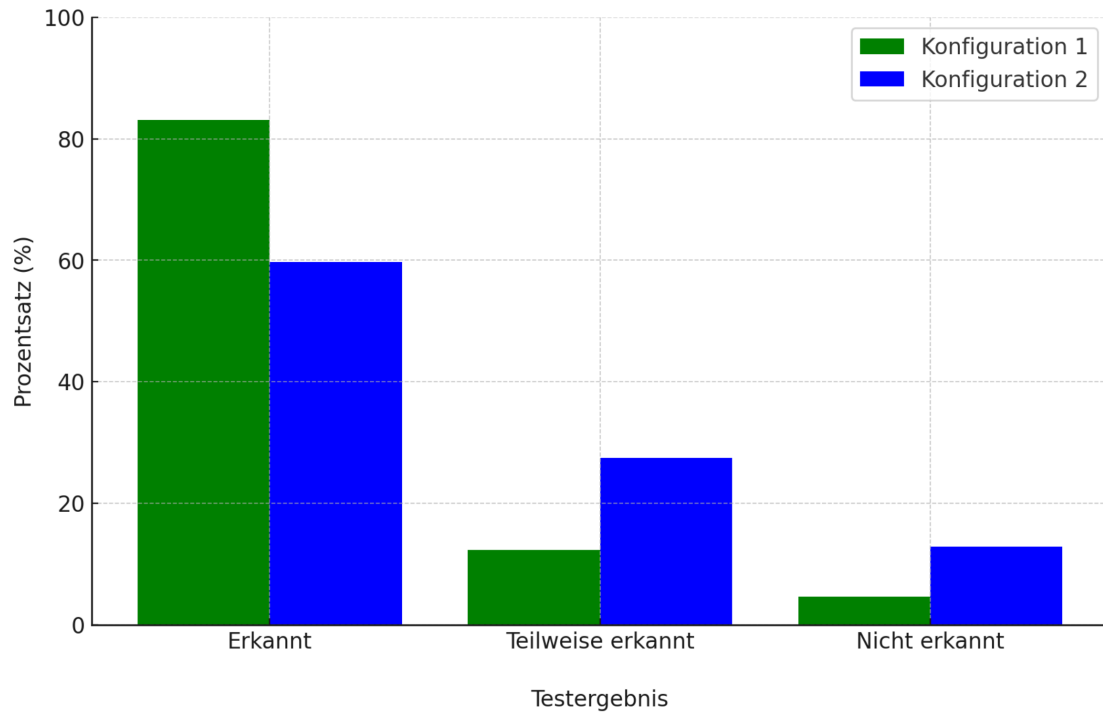


Abb. 10: Testergebnisse für beide Konfigurationen.

Aufgrund der Anordnung auf einer Linie und eines zusätzlichen Ziels in Konfiguration 2 konnten insbesondere die Ziele an den Rändern nicht eindeutig erkannt werden. Eine solche Anordnung ist dennoch möglich, wenn die Anzahl der Ziele auf fünf oder weniger reduziert wird. Dies führt auch bei Zielen an den Rändern der Konfiguration zur besserer Erkennung.

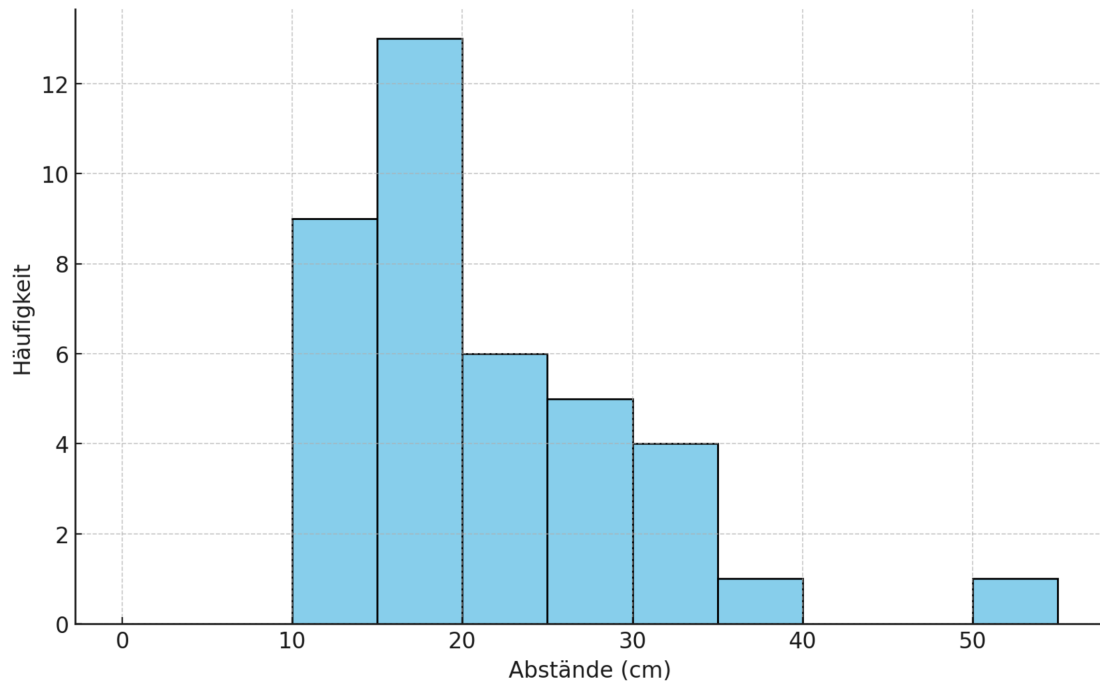


Abb. 11: Häufigkeitsverteilung der Distanzen für Konfiguration 1, bei denen Ziele erfolgreich erkannt wurden.

Abbildung 11 veranschaulicht, wie oft Ziele in verschiedenen Entfernungen erkannt wurden, wobei die durchschnittlichen Entfernung bei etwa 21cm liegt. Die gemessene Distanz ist dabei die erste Stichprobe, in der das Ziel eine Wahrscheinlichkeitsschwelle von 60% erreicht hat.

Im Gegensatz zu generierten Testfällen konnte bei den aufgezeichneten Testfällen durch das Setzen eines minimalen Fortschrittes s_n auf 15% die Wahrscheinlichkeit einer korrekten Erkennung von vielen Zielen deutlich verbessert werden.

Zuletzt konnte durch das Anwenden der Kostenfunktion (Gleichung 8) die Anzahl falsch-positiver Ergebnisse reduziert werden und führte zu konsistenteren Erkennung des Ziels über mehrere aufeinander folgende Stichproben. Grund dafür ist, dass die Gesamtwahrscheinlichkeit aller Ziele relativ zur Distanz der Handposition abnimmt und insgesamt zwar weniger Ziele die Schwelle von 60% erreichen, aber dafür die Wahrscheinlichkeit für das korrekte Ziel steigt.

4.3 Testen mit Trajektorien aus der Studie

Der letzte Test befasst sich mit den in der Studie erfassten Daten von Handtrajektorien. Diese wurden größtenteils mit einer Hololens in 60 Hz und alternativ mit einer Kinect in 10 Hz aufgenommen. Auch wenn diese Daten im Fokus einer anderen Aufgabe entstanden sind und nicht für die Validierung meiner Implementierung vorgesehen waren, wurden sie dennoch als zusätzliches Mittel genutzt, um die Implementierung weiter zu testen. Insbesondere wurden damit die Grenzen des Ansatzes in Bezug auf eine ungeeignet hohe Anzahl sowie die Positionierung der betrachteten Ziele ausgewertet.

4.3.1 Aufbau der Studie und Ablauf des Tests

Im Rahmen der Mensch-Roboter-Zusammenarbeit wurde mit einer Nutzerstudie ein Fabrikszenario simuliert, in dem der Teilnehmer zusammen mit einem Roboter eine Struktur aus Blöcken aufgestellt und dann wieder abgebaut haben.

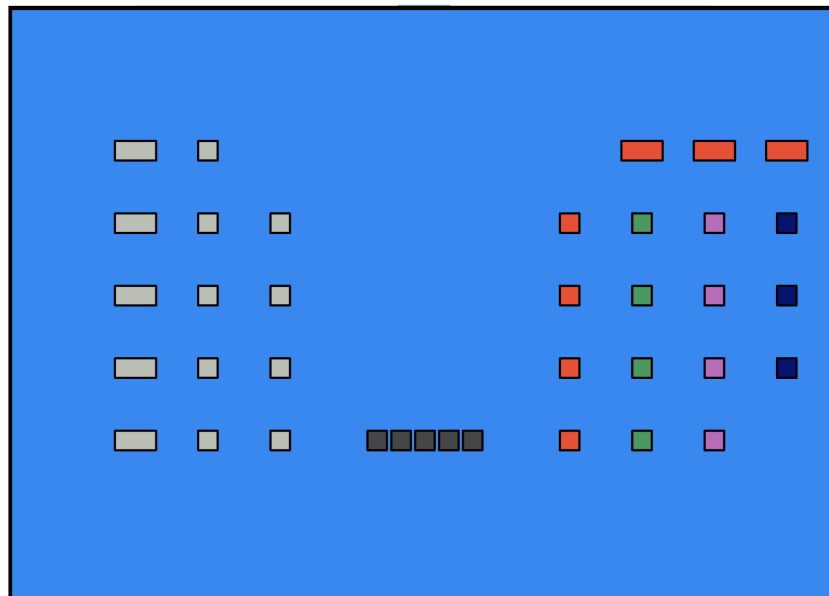


Abb. 12: Tisch auf dem bunte und weiße Blöcken als Zielobjekte liegen, mit Ablageort für die Struktur in grau. (Quelle: unveröffentlichtes Material [Höllerich24])

Abbildung 12 zeigt die Positionierung der Ziele, die in der Studie verwendet wurden. Mit einem Roboter (*oben*) der gegenüber dem Teilnehmer (*unten*) montiert ist. Zusammen wurde eine von zwei Strukturen (Abbildung 13) auf dem dunkelgrauen Ablageort in der Mitte aufgebaut, die aus weißen Blöcken (*links*) und bunten Blöcken (*rechts*) besteht. Während der Bearbeitung dieser Aufgaben wurden dem Teilnehmer die Handlungen des



Abb. 13: Links Struktur 1 und rechts Struktur 2, die aufgebaut wurden.
(Quelle: unveröffentlichtes Material [Höllerich24])

Roboters über die Hololens durch *Augmented Reality* mitgeteilt. Die Aktionen des Roboters basieren dabei auf dem aktuellen Zustand der Struktur und Auswahl der übrigen Blöcke auf dem Tisch. Zusätzlich berücksichtigt der Roboter die Handlungen des Teilnehmers, indem er Blöcke auswählt, die der Teilnehmer voraussichtlich nicht auswählen wird¹⁵.

Der Test mit diesen Daten besteht aus zwei Teilen: Es wurden jeweils fünf Trajektorien für die zwei Aufbaumöglichkeiten der Struktur herangezogen und mit allen Blöcken¹⁶ bzw. Zielen auf der Matte durchgeführt. Im zweiten Teil wurde dann der Abbau aus diesen Blöcken betrachtet, reduziert auf die Blöcke, die im Aufbau der Struktur verwendet wurden.

4.3.2 Ergebnisse

Die Ergebnisse bestätigten erneut, dass die Anzahl der Ziele sowie deren Positionierung entscheidend sind dafür, wie gut die Zielerkennung funktioniert. Die hohe Anzahl der Ziele und die unterschiedlichen Distanzen zur Hand führten zusätzlich dazu, dass viele nicht angestrebte Ziele erkannt wurden. Abbildung 14 zeigt sehr deutlich, wie sehr sich die erfolgreiche Zielerkennung mit der Anzahl steigender Ziele verändert.

¹⁵ Dies ist nur eine von drei Einstellungen des Roboters. In der Nutzerstudie gab es zusätzlich Durchläufe, in denen der Roboter die Blöcke von links nach rechts auswählte oder die Auswahl des Teilnehmers nachahmte.

¹⁶ Um die Anzahl der Ziele zu reduzieren, wurden während der Tests meiner Implementierung die dunkelblauen Blöcke nicht betrachtet, da diese nicht verwendet wurden.

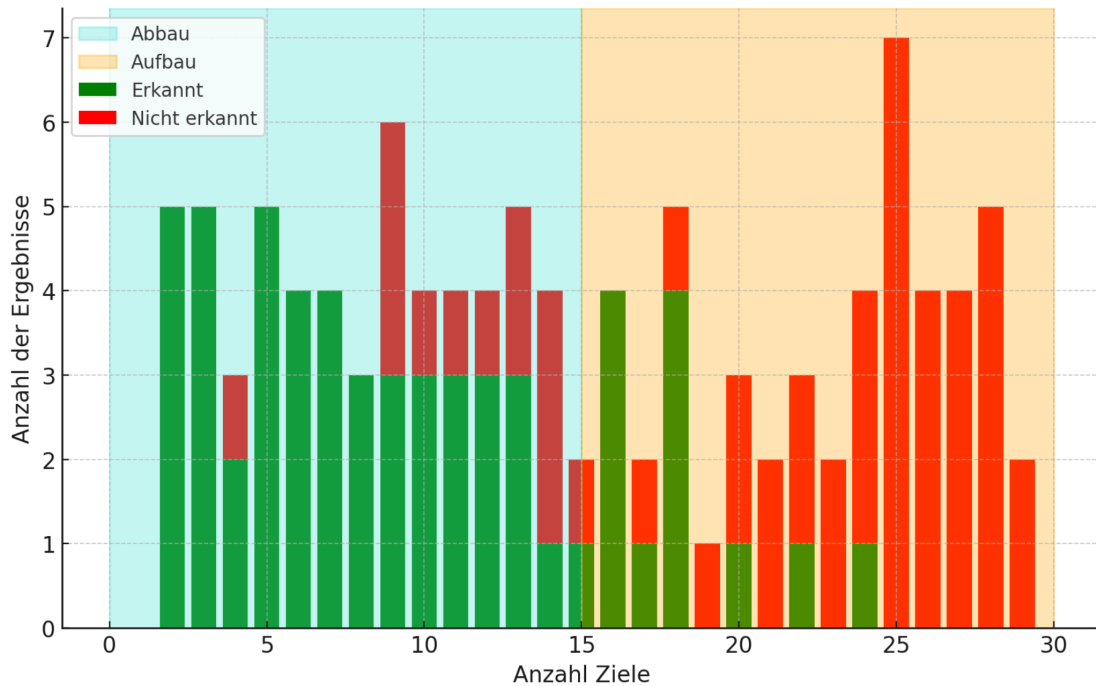


Abb. 14: Häufigkeitsverteilung der Distanzen, bei denen Ziele erfolgreich erkannt wurden.

Dabei sind die meisten Daten aus der ersten Hälfte aus dem Abbau der Struktur entstanden und die Ergebnisse ab 15 betrachteter Ziele aus dem Aufbau. Da im Abbau nur die Ziele betrachtet worden sind, die im Aufbau davor verwendet wurden, gab es da eine bessere Verteilung mit einer geringeren Anzahl an Zielen, was wiederum zu einer besseren Erkennung geführt hat. Es wurden in diesem Fall nur die Ergebnisse *erkannt* und *nicht erkannt* betrachtet, da diese verdeutlichen, ab welcher Anzahl von Zielen die Erkennung sich verschlechtert. Die teilweise Erkennung konnte über alle Anzahlen der Ziele beobachtet werden, aber aufgrund der Vielzahl sowie Positionierung der Ziele konnten diese größtenteils nicht eindeutig eingeordnet werden und waren dementsprechend nicht relevant, um Schlussfolgerungen daraus zu ziehen.

Bezüglich der Teilnehmer und der Betrachtung der jeweiligen Hand gab es folgende Beobachtungen: Die Wahrscheinlichkeit einer erfolgreichen Zielerkennung erhöhte sich, wenn der Teilnehmer Ziele auswählte, die sich in unmittelbarer Nähe zur Hand befanden. Die erfolgreiche Zielerkennung verringerte sich, wenn zwischen der Hand und dem angestrebten Ziel weitere Blöcke lagen. Im Grunde gab es keine großen Unterschiede zwischen den Teilnehmern und der Hand, mit dem diese Blöcke aufgehoben wurden, sondern mit der Reihenfolge und Auswahl der Ziele, die eine erfolgreiche Zielerkennung ausgemacht haben. So kann beispielsweise in Abbildung 14 beobachten, dass auch bei einer Anzahl von 16 oder 18 Zielen korrekt vorhersagt wurde.

4.3.3 Verbesserungsmöglichkeiten der Vorhersage am Beispiel der Studie

Eine Gruppierung aller gleichartigen Blöcke auf sieben einzelne Zielpositionen mit der Konfiguration 1 aus Abbildung 9 würde die Erfolgsrate erheblich verbessern. Der Nachteil wäre dabei lediglich, dass im Fall, dass sowohl Roboter als auch der Mensch auf dieselbe Art von Block zugreifen möchten, einer von beiden entweder kurz warten müsste oder alternativ ein anderer Block für die Bearbeitung im nächsten Schritt ausgewählt werden müsste.

Da die Handlungen des Roboters auf dem Zustand der Umwelt basierten, wurde dem Teilnehmer über die Hololens mitgeteilt, wohin sich der Roboter als Nächstes bewegen würde. Der Teilnehmer musste sich somit an die Handlungen des Roboters richten, um effizient und sicher zu arbeiten. Der Roboter berücksichtigte die Handlungen des Menschen, indem er Blöcke auswählte, die der Teilnehmer voraussichtlich nicht nehmen würde. Durch die Anwendung dieser Intentionsvorhersage könnte der Roboter jedoch zusätzlich proaktiv auf die Handlungen des Teilnehmers reagieren, um die parallele Bearbeitung zu verbessern und eine zusätzliche Sicherheitsmaßnahme zu bilden.

4.4 Zusammenfassung der Ergebnisse

Zuletzt werden alle Ergebnisse aus den drei Tests zusammengefasst. Bei der **Zielkonfiguration** für die Positionierung der Ziele sollten folgende Punkte beachtet werden:

- **Verdeckung:** Es sollte vermieden werden, Ziele so zu positionieren, dass zwischen der Hand und dem angestrebten Ziel ein weiteres Ziel stehen könnte, da dies zu falscher Vorhersage führen kann.
- **Abstand:** Bei der Positionierung sollten die Ziele den gleichen Abstand zueinander haben und am besten auch den gleichen Abstand von den Zielen zu der Basis, von der die Hand am meisten Ziele anstreben wird.
- **Anzahl der Ziele:** Je höher die Anzahl der Ziele ist, desto höher ist die Anzahl falsch-positiver Ergebnisse und desto geringer die zuverlässige Vorhersage des korrekten Zieles. Mit einer Anzahl von 7 Zielen konnten die besten Ergebnisse festgestellt werden.

Eine gute Vorhersage ist neben der Zielkonfiguration auch von weiteren Faktoren abhängig. In erster Linie steht dabei die präzise Erfassung der Handgelenkkoordinaten. Da der Algorithmus sehr stark gegen Rauschen reagiert, sollte einerseits die Messung so präzise wie möglich sein und des Weiteren eine Strategie zur Glättung der Daten verwendet werden. Da die Wahrscheinlichkeitsrechnung auf der Basis von aufeinander folgenden Messwerten berechnet wird, ist die Anzahl der Messungen entlang der Trajektorie ein weiterer entscheidender Punkt. Die Frequenz für die Abtastung sollte hoch genug sein, dass auch schnelle Handbewegungen genug Stichproben liefern, sodass eine zuverlässige Vorhersage möglich ist. Schließlich kann die Kostenfunktion benutzt werden, um die Anzahl der falsch-positiven Ergebnisse zu reduzieren¹⁷.

¹⁷ **Anmerkung:** Bei den aufgezeichneten Tests 4.2 und 4.3 wird bei einer Distanz zwischen zwei Messungen, die kleiner als 1 cm ist, keine Vorhersage modelliert, aber die Wahrscheinlichkeit dennoch zur vorherigen Messung berechnet. Bei den generierten Tests 4.1 hat die minimale Distanz von 2.5 cm zu besseren Ergebnissen geführt.

5 Abschluss

5.1 Industrielle Anwendungsmöglichkeiten

Mit den Stärken dieses Verfahrens sowie den Ergebnissen aus Kapitel 4 ist es durchaus möglich, eine gute Anwendungsmöglichkeit im industriellen Kontext zu finden, in der sich Roboter und Mensch einen gemeinsamen Arbeitsraum teilen. Ein geringer Aufwand in der Vorbereitung sowie universelle Anwendbarkeit machen dieses Verfahren besonders attraktiv, um die Zusammenarbeit mit dem Roboter effizienter zu gestalten und bilden zudem, durch die frühzeitige Erkennung der Ziele, eine weitere Sicherheitsmaßnahme im Umgang mit dem Roboter.

5.2 Mögliche Weiterentwicklungen

Es gibt drei mögliche Weiterentwicklungen, die dieses Verfahren noch weiter verbessern könnten und nicht in dieser Arbeit verfolgt wurden.

Betrachtung der Z-Komponente: Obwohl die Trajektorien im dreidimensionalen Raum modelliert sind, wird die Z-Komponente bei der Berechnung der Wahrscheinlichkeiten nicht direkt verwendet. Eine mögliche Erweiterung wäre, auch die Richtungsvektoren in der XZ- und YZ-Ebene in die Berechnung einfließen zu lassen, um die Erkennung der Ziele weiterhin zu verbessern. So könnte beispielsweise eine Bewegung nach unten, zum Greifen eines Zieles, in Betracht gezogen werden.

Kontextuelle Hinweise: In Kapitel 2 wurde kurz auf die Bedeutung der kontextuellen Hinweise eingegangen. Auch hier könnten Methoden eingesetzt werden, um weiter die Zielerkennung zu verbessern. Beispielsweise wäre denkbar, nur Ziele zu betrachten, die im Blickwinkel der Person liegen, um die Anzahl der Ziele zu begrenzen und somit die Wahrscheinlichkeit der betrachteten Ziele zu erhöhen.

Zunahme weiterer Messungen zur Modellierung: Die Berechnung der vorhergesagten Trajektorien aus nur zwei Messungen ermöglicht zwar eine schnelle Reaktion auf eine Umentscheidung auf ein anderes Ziel, macht es jedoch anfällig für Messfehler. Die Hinzunahme von zusätzlichen Messungen könnte dazu führen, dass aufeinanderfolgende Trajektorien konsistenter wären, was zu einer gleichmäßigeren Anpassung der Wahrscheinlichkeiten führen würde.

5.3 Zusammenfassung

Das Eintreten des Roboters in den Arbeitsraum des Menschen stellt viele Herausforderungen dar. Um die Sicherheit des Menschen zu gewährleisten und die Zusammenarbeit mit dem Roboter zu optimieren, spielt die Vorhersage menschlicher Bewegungen eine wesentliche Rolle. Zu diesem Zweck wurden viele Forschungen durchgeführt und Ansätze angefertigt, wobei jedes von ihnen ihre eigenen Stärken und Schwächen aufweisen. Der Ansatz von Zanchettin und Rocco verspricht dabei mit wenigen Daten eine frühzeitige und konsistente Vorhersage menschlicher Handbewegungen, wodurch sich neue Möglichkeiten in der industriellen Anwendung eröffnen. Dazu wird zu jedem Ziel eine Vorhersage über die zukünftige Handposition modelliert und mit aktuellen Messungen des Handgelenks verglichen, worauf schließlich die Wahrscheinlichkeit für diese Ziele berechnet wird. Dieses Verfahren wurde in Python implementiert und dessen Funktionalität durch zahlreiche Tests bestätigt. Weiterhin konnte durch diese Tests das Verfahren weiter optimiert werden und passende Konfigurationen ermittelt werden. Eine gleichmäßige Anordnung von sieben Zielen um die Basis, hat sich dabei als beste Positionierung erwiesen. Die Ergebnisse zeigen, dass durch eine optimale Positionierung erfolgreiche Vorhersagen menschlicher Handbewegungen ermöglicht werden. Insgesamt bietet dieser Ansatz vielversprechende Möglichkeiten für eine sichere und effiziente Mensch-Roboter-Kollaboration, die auf verschiedene industrielle Szenarien angewendet werden können.

Literatur

- [Castro21] Castro A., Silva F., Santos V.: “Trends of Human-Robot Collaboration in Industry Contexts: Handover, Learning, and Metrics”, *Sensors*, Band 21, Nr. 12, 2021.
- [Formica21] Formica F., Vaghi S., Lucci N., Zanchettin A. M.: “Neural Networks based Human Intent Prediction for Collaborative Robotics Applications”, In 2021 20th International Conference on Advanced Robotics (ICAR), 2021.
- [Hu16] Hu B., Yuan J., Wu Y.: “Discriminative Action States Discovery for Online Action Recognition”, *IEEE Signal Processing Letters*, Band 23, Nr. 10, 2016.
- [Höllerich24] Höllerich N.: “Fluency in dynamic human-robot teaming with intention prediction”, Doktorarbeit, Universität Bayreuth, Lehrstuhl für Robotik und Eingebettete Systeme, 2024, Unveröffentlichte Dissertation, Datum der Veröffentlichung noch nicht bekannt.
- [Hübner20] Hübner P., Clintworth K., Liu Q., Weinmann M., Wurstthorn S.: “Evaluation of HoloLens Tracking and Depth Sensing for Indoor Mapping Applications”, *Sensors*, Band 20, Nr. 4, 2020.
- [Khan23] Khan F., Asif S., Webb P.: “Communication components for Human Intention Prediction – A Survey”, In Karwowski W., Trzcielinski S. (Hg.), *Human Aspects of Advanced Manufacturing. AHFE International Conference*, Band 80, 2023.
- [Kong22] Kong Y., Fu Y.: “Human action recognition and prediction: A survey”, *International Journal of Computer Vision*, Band 130, Nr. 5, Seiten 1366–1401, 2022.
- [Lotysh23] Lotysh V., Gumeniuk L., Humeniuk P.: “COMPARISON OF THE EFFECTIVENESS OF TIME SERIES ANALYSIS METHODS: SMA, WMA, EMA, EWMA, AND KALMAN FILTER FOR DATA ANALYSIS”, *Informatyka, Automatyka, Pomiary w Gospodarce i Ochronie Środowiska*, Band 13, Nr. 3, 2023.
- [Mainprice15] Mainprice J., Hayne R., Berenson D.: “Predicting human reaching motion in collaborative tasks using Inverse Optimal Control and iterative re-planning”, In 2015 IEEE International Conference on Robotics and Automation (ICRA), 2015.
- [PAR11] PAR: “Wrapped normal distribution — Wikipedia, The Free Encyclopedia”: <http://en.wikipedia.org/w/index.php?title=Wrapped%20normal%20distribution&oldid=1179272281>, 2011, [Online; accessed 18-August-2024].

- [Peifer24] Peifer Y.: Konzeptionierung eines arbeitswissenschaftlichen handlungsrahmens, Seiten 199–252, Springer Berlin Heidelberg, Berlin, Heidelberg, 2024, ISBN: 978-3-662-68560-0.
- [Pérez-D’Arpino15] Pérez-D’Arpino C., Shah J. A.: “Fast target prediction of human reaching motion for cooperative human-robot manipulation tasks using time series classification”, In 2015 IEEE International Conference on Robotics and Automation (ICRA), 2015.
- [Rudenko20] Rudenko A., Palmieri L., Herman M., Kitani K. M., Gavrila D. M., Arras K. O.: “Human motion trajectory prediction: a survey”, The International Journal of Robotics Research, Band 39, Nr. 8, Seiten 895–935, 2020.
- [Subetha16] Subetha T., Chitrakala S.: “A survey on human activity recognition from videos”, In 2016 International Conference on Information Communication and Embedded Systems (ICICES), 2016.
- [Zanchettin17] Zanchettin A. M., Rocco P.: “Probabilistic inference of human arm reaching target for effective human-robot collaboration”, In 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2017.

Abbildungsverzeichnis

1	Überblick über die verschiedenen Ansätze der Bewegungsvorhersage aus [Rudenko20].	9
2	Beispiel der Modellierung von zwei vorhergesagten Trajektorien zu den jeweiligen Zielen in der XY-Ebene aus [Zanchettin17].	15
3	Wahrscheinlichkeitsdichtefunktion mit Mittelwert $\mu = 0$ und mehreren Varianzen σ^2 im Vergleich aus [PAR11].	16
4	Beispiel aus [Zanchettin17], in der zwei Ziele durch ähnliche Richtungsvektoren fälschlicherweise die gleiche Wahrscheinlichkeit haben.	18
5	Zielkonfigurationen für die Trajektorien generiert wurden.	21
6	Skizze generierte Trajektorien zu einem Ziel.	22
7	Erfolgreiche Zielerkennung in Abhängigkeit von der unteren Grenze der Varianz σ^2	23
8	Fall, in dem sich zwei vorhergesagte Trajektorien zu Ziel 1 und Ziel 2 überdecken.	24
9	Links: Skizze der 1. Konfiguration; Rechts: Skizze der 2. Konfiguration . .	25
10	Testergebnisse für beide Konfigurationen.	26
11	Häufigkeitsverteilung der Distanzen für Konfiguration 1, bei denen Ziele erfolgreich erkannt wurden.	27
12	Tisch auf dem bunte und weiße Blöcken als Zielobjekte liegen, mit Abgelegt für die Struktur in grau. (Quelle: unveröffentlichtes Material [Höllerich24])	28
13	Links Struktur 1 und rechts Struktur 2, die aufgebaut wurden. (Quelle: unveröffentlichtes Material [Höllerich24])	29
14	Häufigkeitsverteilung der Distanzen, bei denen Ziele erfolgreich erkannt wurden.	30