Bei diesem Beitrag handelt es sich um einen wissenschaftlich begutachteten und freigegebenen Fachaufsatz („reviewed paper“).

doi.org/10.37544/1436-4980-2024-09-67

Kosteneffiziente sensorgestützte Montage mit Robotern

Montagemanipulator für die THT-Bestückung

M. Polikarpov, Y. Mehmood, D. Boiar, J. Deuse

ZUSAMMENFASSUNG In diesem Beitrag wird die Anwendung des parallelkinematischen Manipulators Hexaglide in der THT-Bestückung untersucht. Kostengünstige optische und taktile Sensoren ermöglichen präzises Referenzieren und Einstecken von Bauteilen. Eine kraft- und bildgeregelte heuristische Steuerungsstrategie sowie Reinforcement Learning wurden erprobt und die Tauglichkeit des Hexaglide für die Bestückung wurde experimentell gezeigt.

STICHWÖRTER

Automatisierung, Mechatronik, Montage

Cost-efficient sensor-assisted robotic assembly – Assembly manipulator for PCB assembly

ABSTRACT In this article, the Hexaglide manipulator, a parallel robot, is used for automating Printed Circuit Boards (PCB) assembly of through-hole components. Low-cost optical and tactile sensors facilitate accurate component referencing and part insertion. A heuristic algorithm based on image-based visual servoing and force control as well as Reinforcement Learning are demonstrated. Experimental results validate the effectiveness of Hexaglide for PCB assembly automation.

1 Einleitung

Leiterplatten sind ein wesentlicher Bestandteil von elektronischen Produkten. Elektronische Bauelemente werden bei der Bestückung entsprechend einem definierten Muster auf der Leiterplatte platziert und stoffschlüssig befestigt. Neben den mehr verbreiteten Methoden der Oberflächenmontage (engl. Surface Mount Technology, SMT) findet die Durchsteckmontage (engl. Through Hole Technology, THT) Anwendung in anspruchsvollen Produkten [1]. Die Durchsteckmontage wird in der Herstellung der Leistungselektronik aufgrund hoher Verbindungsfestigkeit und thermischer Beständigkeit eingesetzt [2, 3]. THT-Bauteile besitzen im Allgemeinen zwei oder mehr Anschlussdrähte, die in entsprechende Löcher auf der Leiterplatte eingesteckt werden. Dabei müssen die Bauteile am richtigen Ort beschädigungsfrei montiert werden, auch wenn ihre Anschlussdrähte (Pins) für Biegung empfindlich sind und die Fügetoleranzen nur wenige Zehntelmillimeter betragen. Aufgrund dieser Eigenschaften der Bestückungsaufgabe ist die THT-Montage ein zeitintensiver, nicht immer wirtschaftlich zu automatisierender, Prozess und kann zwischen 40 % und 70 % der gesamten Produktionszeit in der Elektronikfertigung beanspruchen [1, 4].

In der Forschung werden unterschiedliche Ansätze verfolgt, um die THT-Bestückung mithilfe von Robotern zu automatisieren. Roboterbasierte Systeme weisen im Allgemeinen eine höhere Flexibilität auf als spezialisierte Bestückungsautomaten und können bei Bedarf auch in anderen Produktionsaufgaben eingesetzt werden. Das Fügen ist eine der zentralen Herausforderungen für die robotergestützte THT-Bestückung. In der Regel besitzen marktübliche Leichtbauroboter nicht die erforderliche Absolutgenauigkeit, um mit engen Fügetoleranzen und ohne zusätzliche Kalibrierung umgehen zu können. Entweder muss das Robotersystem und die Fördertechnik dafür mit 3D-Kameras präzise vermessen werden [5] oder der Roboter muss mit Kameras [6–8] oder einem Kraftmomentensensor [7] ausgestattet werden. Damit wird der Roboter in die Lage versetzt, anhand der Sensorsignale und geeigneter Suchstrategien die Löcher in der Leiterplatte durch Antasten zu finden. In [9] wurde eine Methode zur datengetriebenen Optimierung von kraftgeregelten Suchstrategien vorgestellt, um die Suchzeit zu verkürzen. Der Ansatz des bestärkenden Lernens (engl. Reinforcement Learning, RL) in der THT-Bestückung anhand von Kraft- und Bilddaten wurde von *Bartyzel et al.* [10] untersucht.

Obwohl unterschiedliche Ansätze vielversprechende Erfolgsquoten bei der sensorgestützten roboterbasierten THT-Bestückung erzielen, sind diese auf teure Hardware, auf große (Trainings-) Datenmengen oder auf sehr beschränke Anwendungsfälle zurückzuführen. In diesem Beitrag wird ein neuer kostengünstiger Hexaglide-Manipulator vorgestellt, der hohe Bewegungspräzision für Einsteckvorgänge aufweist (Abschnitt 2). Im Abschnitt 3 werden zwei untersuchte Kategorien von Steuerungsansätzen erläutert. Darauffolgend werden die Erkenntnisse aus den experimentellen Untersuchungen mit dem Hexaglide im Abschnitt 4 betrachtet und abschließend wird das Gesamtergebnis im Abschnitt 5 diskutiert.

2 Hexaglide-Manipulator

Der im Forschungsprojekt „SmartAssembly“ entstandene Prototyp „Hexaglide“ (**Bild 1**) ist ein Manipulatorsystem, das die Positionierung der zu bestückenden Leiterplatte relativ zum Bauteil übernimmt. Das THT-Bauteil wird durch eine beliebige geeig-

https://doi.org/10.37544/1436-4980-2024-09-67 https://www.inlibra.com/de/page/agb - Open Access - (cc) BY

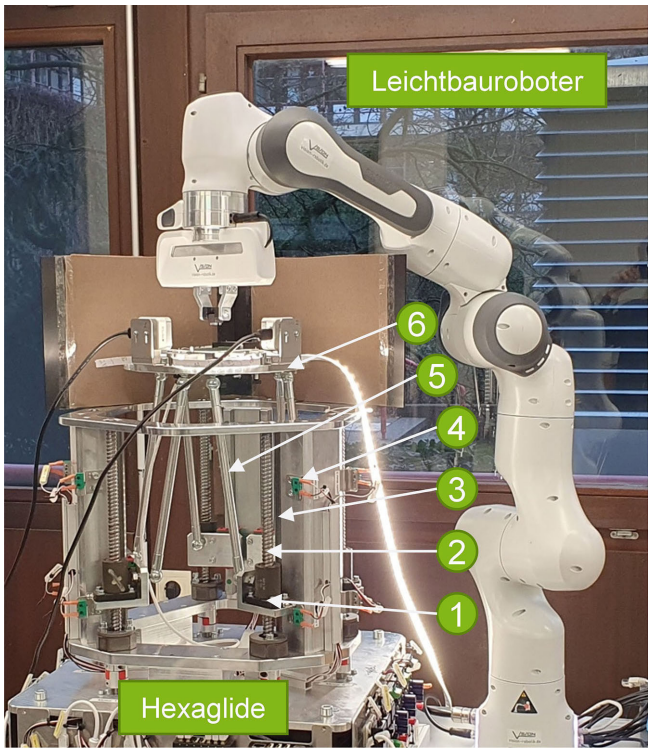

Bild 1. Montagesystem bestehend aus dem Hexaglide und einem Leichtbauroboter: 1 – Spindelwinkel, 2 – Gewindespindel, 3 – Gleitschiene, 4 – Endschalter, 5 – Stange mit Kugelgelenken, 6 – Plattform. *Foto: IPS, SFA GmbH & Co KG*

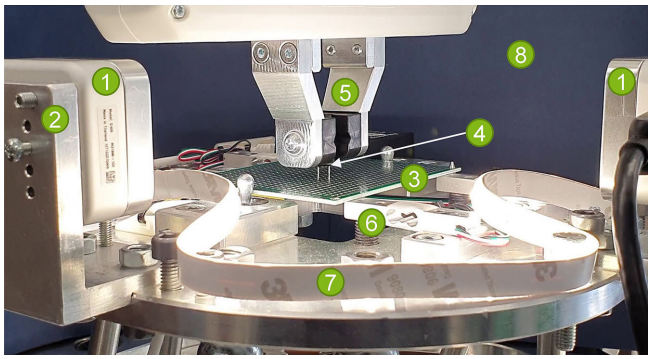

Bild 2. Plattform für die Aufnahme von Leiterplatten. 1 – Kameras, 2 – Halterung, 3 – Leiterplatte, 4 – THT-Bauteil, 5 – Greifer, 6 – Dehnmessstreifen, 7 – LED, 8 – Hintergrund. *Foto: IPS, SFA GmbH & Co KG*

nete Vorrichtung oder durch einen Leichtbauroboter unbeweglich gehalten. Damit kann die erforderliche Präzision der Relativbewegung zwischen der Leiterplatte und dem Bauteil unabhängig von Leichtbaurobotermodellen reproduzierbar erreicht werden. Das unterscheidet sich von den in der Literatur beschriebenen Ansätzen [5, 8–11] und Anwendungsfällen in der Elektronikfertigung, in denen Leichtbauroboter die gesamte Fügebewegung ausführen und die Leiterplatte bei der Montage auf einem Förderband statisch positioniert ist [12–14]. Im Folgenden werden Systembestandteile des entwickelten Hexaglide und maßgebliche Designentscheidungen im Einzelnen beschrieben.

2.1 Mechanischer Aufbau und kinematische Grundlagen

Der Hexaglide besitzt sechs – in der vorliegenden Ausführung senkrechte – lineare Achsen, deren Bewegung durch die Rotation angetriebener Gewindespindeln erzeugt wird. Auf jeder der Gewindespindeln befinden sich Spindelwinkel, die über Kugelgelenke mit den entsprechenden Stangen verbunden sind. Kugelgelenke auf den oberen Enden der Stangen tragen die Plattform (Endeffektor) für die Aufnahme von Leiterplatten. Zur Sicherstellung der fixierten Orientierung der Spindelwinkel während der Vorschubbewegungen sind senkrechte Gleitschienen vorgesehen. Die Achsenbeschränkungen sind durch taktile Endschalter realisiert. Jede der linearen Vorschubachsen wird durch einen separaten Schrittmotor mithilfe einer Steuereinheit angetrieben. Der beschriebene Aufbau weist alle drei translatorische und drei rotatorische Freiheitsgrade auf. Damit kann der Hexaglide zwei Rotationen mehr als die bei der THT-Bestückung häufig eingesetzten SCARA-Roboter ausführen, sodass auch deutlich verkippte Bauteile montiert werden können (vgl. [6]). Darüber hinaus kann eine bewusste Verkippung bei der Montage von speziellen Bauteilen mit einer großen Anzahl der Pins zielführend sein.

Die Hexaglide-Kinematik gehört zur Klasse der geschlossenen kinematischen Ketten. Diese zeichnen sich durch einfache Miniaturisierung, hohe Struktursteifigkeit und hohe Genauigkeit aus [15, 16]. Die Gleichungen aus [17] beschreiben in der allgemeinen Form die Vorwärts- und die Rückwärtskinematik des Hexaglide und wurden für die notwendigen Umrechnungen zwischen Zielposen und Motorbefehlen eingesetzt. Im Gegensatz zu den mehr verbreiteten, offenen kinematischen Ketten (zum Beispiel 6- oder 7-Achs-Knickarmroboter) können die Motorbefehle für gewünschte Zielposen des Hexaglide analytisch, ohne numerische Verfahren bestimmt werden, was für schnelle Berechnungszeiten sorgt. Bei günstiger Maschinengestaltung können Singularitäten und Sprünge in den Gelenkstellungen des Hexaglide vermieden werden.

2.2 Optische und taktile Sensoren

Bei der Steuerung der Bestückung von Leiterplatten müssen Bauteile mit Genauigkeiten im Submillimeterbereich manipuliert werden, um die Pins in die Löcher einzustecken, da die typischen Spiele zwischen dem Loch und Pin 0,1–0,5 mm betragen. Aber noch vor dem Einstecken muss die relative räumliche Koordinate des THT-Bauteils zum Bezugskoordinatensystem der Leiterplatte ermittelt werden. Um beide Voraussetzungen kostengünstig und vorzugsweise mit gleichen Sensoren zu erfüllen, wurden im dargestellten Hexaglide-System zwei Kameras mit justierbaren Halterungen auf der Plattform verbaut, **Bild 2**. Die Kameras funktionieren zusammen als eine Stereokamera, um aus einzelnen 2D-Bildern notwendige 3D-Informationen zu gewinnen. Die Einbaulage der Kameras über der Leiterplatte muss so gewählt werden, dass die gesamte Leiterplatte in beiden Kameras sichtbar ist.

Für das im Abschnitt 3.1 beschriebenes Referenzieren des Manipulators zum vorgehaltenen THT-Bauteil ist ein homogener Hintergrund ohne glänzende Reflektionen erforderlich, damit die Pins eindeutig und klar durch die Kameras erkannt werden können. Zur Hervorhebung der reflektierenden Oberflächen der Pins wird ein horizontal angeordneter LED-Streifen mit einem längenbezogenen Lichtstrom von 1000–4000 lm/m verwendet.

https://doi.org/10.37544/1436-4980-2024-09-67 https://www.inlibra.com/de/page/agb - Open Access - CC BY

Bild 3. Steuerungsablauf der intelligenten THT-Bestückung. *Grafik: IPS*

Der Hexaglide greift zusätzlich auf Kraftmessungen zurück, um den Fortschritt der Montage zu überwachen (Abschnitt 3.2). Zur Messung von eventuellen Kontaktkräften zwischen dem Bauteil und der Leiterplatte besitzt der Hexaglide vier Dehnmessstreifen mit jeweils einem Messbereich von 7,8 N, die unter den Ecken der Versuchsleiterplatte positioniert sind. Der Messbereich wurde mit einem großen Sicherheitsfaktor dimensioniert, um während des Forschungsprojekts eventuelle starke Kollisionen mit dem haltenden Leichtbauroboter standzuhalten. Die Kraftauflösung von circa 0,01 N erwies sich als ausreichend für die Erkennung wesentlicher Zustände während der Lochsuche und des Einfügens.

2.3 Software und Vernetzung

Als Framework zur Kommunikation und Steuerung der gesamten Aktoren und Sensoren des Manipulators wird ROS Melodic verwendet und auf einem Laborrechner mit Ubuntu 18.04 ausgeführt. Alle Skripte für den Empfang von Sensordaten, ihre anschließende Verarbeitung und die Ausgabe von Motorbefehlen wurden in Python 3.6 implementiert. Die Daten von den Kraftsensoren und von den Motor-Encodern werden über eine USB-Schnittstelle empfangen. Die beiden Intel RealSense Kameras werden auch per USB an den Rechner angeschlossen.

In einer vorausgehenden Publikation [18] wurden die grundsätzlichen Funktionsbestandteile bereits in Tiefe behandelt. Es handelte sich um ein System, in dem ein Leichtbauroboter die Bauteile relativ zu einer fest montierten Leiterplatte bewegte. Die Datenverarbeitung und die Steuerung lassen sich ohne signifikante Anpassungen auf den Hexaglide-Manipulator transferieren, da die Kameratechnik gleichgeblieben ist und die Bewegungsbefehle stets in kartesischen Koordinaten bestimmt wurden. Damit ist die Logik der Steuerung größtenteils von der verwendeten Roboterkinematik unabhängig und zwischen verschiedenen Robotersystemen übertragbar.

Im nächsten Abschnitt werden die für die Steuerung des Hexaglide ausgearbeiteten Algorithmen und Methoden der sensorgestützten Montage erläutert, um besseres Verständnis der experimentellen Untersuchungen zu ermöglichen.

3 Steuerungsalgorithmen

Analog zum Ablauf der THT-Bestückung, wie diese von Menschen ausgeführt wird, kann auch der Ablauf der robotergestützten THT-Bestückung mithilfe von MTM-Grundbewegungen Hinlangen – Greifen – Bringen – Fügen – Loslassen [19] analysiert werden. Das Hinlangen zu einer Bereitstellungsvorrichtung oder zu einem Behälter sowie das Greifen der THT-Bauteile kann auf viele unterschiedliche Weisen unabhängig vom Hexaglide erfolgen, wobei die konkrete Form der Bauteilbereitstellung stark variieren kann. Je nach betrachtetem Produktionsszenario und -umfeld können Kondensatoren, Widerstände, Relais und Transformatoren in geordneten Magazinen oder Trays bereitgestellt werden oder im schwierigsten Fall als komplett ungeordnetes Schüttgut vorliegen. Der im Fokus des Beitrags stehende Hexaglide-Manipulator erfordert, dass ein Leichtbauroboter oder eine beliebige sonstige Zuführungsvorrichtung in der Lage ist, die Bauteile im Greifer so zu präsentieren, dass die Pins senkrecht orientiert sind. Wenn das erfolgt ist, übernimmt der Hexaglide das Bringen und das Fügen des THT-Bauteils, indem die Leiterplatte bewegt wird. **Bild 3** veranschaulicht den Bewegungsablauf des Hexaglide und die erforderlichen Schritte der Steuerung.

3.1 Referenzieren des Manipulators zum Bauteil

Als Erstes muss die relative Position und Orientierung des Bauteils zur Leiterplatte bestimmt werden. Als wesentliches und einfach zu detektierendes Merkmal eines THT-Bauteils können die Spitzen der Pins verwendet werden. Wie in [18] gezeigt wurde, reichen für die Lokalisierung der Pin-Spitzen klassische Methoden der Bildverarbeitung aus. Mithilfe von zwei 2D-Bildaufnahmen, in denen Pin-Spitzen detektiert wurden, können bei bekannten extrinsischen Parametern des Kamerasystems 3D-Koordinaten und die Orientierung der Bauteile im Greifer ermittelt werden. In der ersten Iteration wird der Referenz-Pin festge-

https://doi.org/10.37544/1436-4980-2024-09-67 https://www.inlibra.com/de/page/agb - Open Access - CC BY

legt und zum Schnittpunkt der Hauptachsen der Kameras gebracht. Weiter wird dieser Punkt abgekürzt als optischer Ursprung bezeichnet. Dieser Punkt ist optimal für die nachfolgenden Messungen, da in diesem Punkt die Auflösung beider Kameras gleich ist. Hier wird das Bauteil auf nicht akzeptable Verformungen geprüft, indem der Pin-Abstand vermessen und mit dem Vorgabewert verglichen wird. Anhand erneuter Pin-Spitzen-Lokalisierung wird der vorliegende Orientierungswinkel berechnet und das Bauteil wird automatisch entsprechend dem Layout der Leiterplatte ausgerichtet. Um den Referenzierungsvorgang robust gegenüber Ungenauigkeiten in den Kameraparametern oder äußeren Einflüssen zu machen, wird abschließend eine optische Korrektur der Abweichungen vorgenommen. Ähnlich zur ersten Iteration fährt die Plattform derart, dass sich der Referenz-Pin im optischen Ursprung befindet. Aus dieser Position wird die Anfahrbewegung ausgeführt, die letztendlich die Pins zu den richtigen Löchern bringt.

3.2 Einsteckalgorithmen

Bei der robotergestützten Montage von elektronischen Baugruppen gibt es eine Reihe von Einflüssen auf die Produktivität. Ein besonders fehleranfälliger Schritt ist dabei die unmittelbare Einsteckbewegung. Ein robotisches Bestückungssystem muss in diesem Schritt zahlreiche, durch verschiedene Faktoren hervorgerufene Ungenauigkeiten überwinden:

- Mensch: Umgang mit der Maschine, Eingaben;
- Maschine: Verschleiß, Antrieb, Greifer;
- Milieu: Lichteinflüsse, Vibrationen, Staub;
- Messung: Kalibrierung, Auflösung;
- Methode: Bauteilzufuhr, Steuerung;
- Material: Leiterplatte, Bauteile.

Aus diesem Grund müssen die eingesetzten Einsteckalgorithmen ein hohes Maß an Robustheit gegenüber den Störeinflüssen aufweisen. Im Folgenden werden zwei unterschiedliche Herangehensweisen an diese Aufgabe geschildert: eine Heuristik und Reinforcement Learning.

3.2.1 Heuristische Methode

Im ersten Schritt wird die optische Regelung mithilfe der Pin-Lokalisierung durch Erkennung und spaltenweise Zählung von hell reflektierenden Pixeln vorgenommen [18]. Dafür greift der entwickelte Algorithmus auf Kamerabildausschnitte, die um das vorgegebene Referenz-Loch zentriert sind. Durch die Analyse der spaltenweisen Anzahl von hellen Pixeln kann die horizontale Abweichung des Referenz-Pins ermittelt und durch den Hexaglide korrigiert werden (s. Zentrierung im Bild 3). Erreicht die Abweichung einen zulässigen Toleranzbereich, fährt der Hexaglide in Schritten von 0,5 mm nach oben, um das Bauteil einzustecken. Falls signifikante Kontaktkraft von mehr als 2,5 N auftritt, wird in einem spiralförmigen Muster (diskrete Schritte von 0,5 mm, rechteckig, von innen nach außen, maximaler Radius 1 mm) in der Leiterplattenebene verfahren, bis die Kraft unter 1 N abfällt. Dieser signifikante Kraftabfall deutet auf eine Verbesserung des Montagezustands, das heißt die Löcher wurden durch die kontaktbehaftete Suche gefunden. Ab diesem Moment kann mit der weiteren senkrechten Einsteckbewegung fortgefahren werden. Im Fall einer komplett erfolglosen Spiralsuche wird die optische Zentrierung des Referenz-Pins wiederholt vorgenommen und die Einsteckbewegung wird wieder eingeleitet. Auf Grundlage von Versuchen mit verschiedenen Elektronikbauteilen wurden 7 N als obere zulässige Kontaktkraft zwischen Pins und Leiterplatte festgesetzt, da ab dieser Kraft die Pins tendenziell plastisch verformen und knicken. Eine erfolgreiche Einsteckbewegung wird ab einer Einstecktiefe von mehr als 2 mm aufgezeichnet. Die genannten Parameter wurden empirisch während der Entwicklung ermittelt, können aber je nach Anforderungen in Konfigurationsdateien angepasst werden.

3.2.2 Reinforcement Learning

Reinforcement Learning (RL) ist eine grundlegend unterschiedliche Methode, die ohne menschliches Zutun die Handlungsstrategie erlernt. Dabei wird in den aktuellen Ansätzen aus der Forschung dafür in der Regel ein Black-Box-Modell wie neuronales Netz durch die Trainingserfahrungen angepasst, um für eine Beobachtung die zielführendste Aktion auszuwählen. Mathematisch ausgedrückt, wird das Ziel verfolgt, die Summe der Belohnungen über die Zeit zu maximieren. Der Algorithmus (Agent) lernt die Handlungsstrategie durch die Interaktion mit der Umgebung in mehreren Versuchen (Episoden), wobei die ersten Episoden anhand eines zufällig initialisierten Modells ausgeführt werden. [20]

Für die Anwendung von RL muss als Erstes definiert werden, welche Beobachtungen (States) aus der Umgebung für den Algorithmus relevant sind. Beim Einstecken der THT-Bauteile mit Robotern kann in Anlehnung an die menschliche Wahrnehmung auf Bild-, Positions- und Kraftdaten zurückgegriffen werden, **Bild 4**. Um die Dynamik des Einsteckvorgangs zu berücksichtigen, werden Bilder aus dem aktuellen und aus dem vorigen Zeitschritt in die neuronalen Netze eingespeist. Im Gegensatz zu den Bildausschnitten, die bei der heuristischen Methode Einsatz finden, soll das RL-Modell anhand der Aufnahmen des ganzen Bauteils über nächste Handlungen entscheiden. Dadurch soll erforscht werden, ob ein Black-Box-Modell in der Lage ist, deutlich komplexere Bilddaten als bei der Heuristik zu interpretieren. Dafür werden mit einer dritten, extern montierten Kamera Bilder mit einer Größe von 256x256 Pixel aufgenommen, zentriert um das Bauteil. Da auch diese vergleichsweise kleinen Bilder mit drei Farbkanälen eine sehr hohe Dimensionalität haben (196 608 Zahlen pro Bild), ist es sinnvoll, die Bilddaten auf einen Vektor zielführender Merkmale zu reduzieren. Dafür sind Methoden des Transfer Learning geeignet [21]. In unserem System wird auf jedem Bild ein auf ImageNet-Datensatz [22] vortrainiertes Faltungsnetz ResNet50 [23] angewendet. Das ermöglicht eine Reduktion der gesamten multimodalen Beobachtung auf lediglich 136 Merkmale. Diese komprimierte Beobachtung wird nachfolgend in drei Stufen durch vollvernetzte Neuronenschichten auf sechs Ausgabeneuronen zusammengeführt. Diese entsprechen den Aktionen, die der Algorithmus mit dem Hexaglide ausführen kann (jeweils 0,5 mm translatorische Schritte in sechs möglichen Richtungen). Um die Trainingsdauer zu reduzieren und die generelle Machbarkeit schneller zu untersuchen, wurde auf Rotationen als mögliche Aktionen verzichtet.

Um zwischen günstigen und ungünstigen Handlungen zu unterscheiden, wird eine Belohnungsfunktion benötigt, die dem Algorithmus entsprechend Plus- oder Minuspunkte vergibt. Während der Anlaufversuche konnte eine zusammengesetzte Belohnungsfunktion entworfen werden, die die Kontaktkraft (r_f), die

https://doi.org/10.37544/1436-4980-2024-09-67 https://www.inlibra.com/de/page/agb - Open Access - CC BY

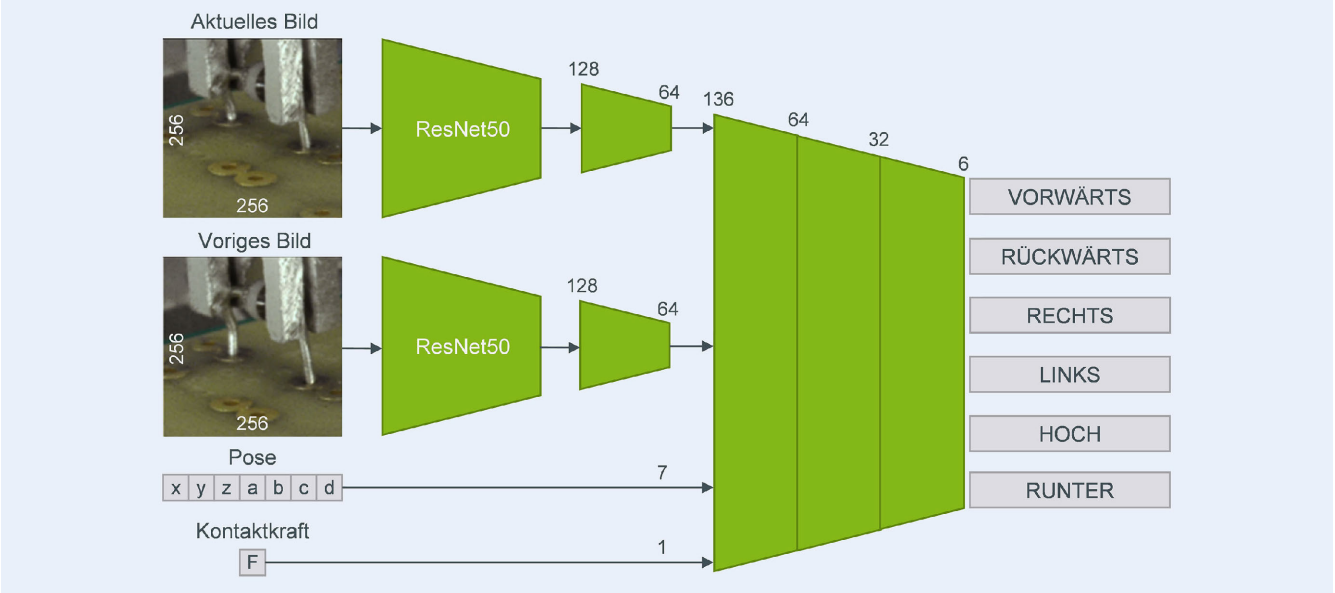

Bild 4. Architektur der Datenverarbeitung für RL. *Grafik: IPS*

Einstecktiefe (r_d), die Bewegungsdauer (r_t), die Überschreitung der sinnvollen Arbeitsraumgrenzen (r_w) sowie das finale Erfolgssignal (r_s) in jedem Zeitschritt berücksichtigt:

$$r = r_f + r_d + r_t + r_w + r_s$$

Kontaktkräfte unter 2 N werden in der Belohnungsfunktion nicht berücksichtigt. Übersteigt die Kontaktkraft 2 N, wird jeder überflüssige 1 N mit −5 Punkten bestraft, damit zu große Krafteinwirkung auf die Pins vermieden wird:

$$r_f = \begin{cases} 0, & falls\ F_z < 2\ N \\ -5 \cdot (F_z - 2\ N), & sonst \end{cases}$$

Wenn die Einstecktiefe steigt, bekommt der Algorithmus 20 Punkte, und wird mit −20 bestraft, wenn die Plattform vom Bauteil wegfährt, damit der Algorithmus motiviert ist, das Bauteil in die Leiterplatte einzustecken:

$$r_d = \begin{cases} 20, & falls\ d_t > d_{t-1} \\ 0, & falls\ d_t = d_{t-1} \\ -20, & falls\ d_t < d_{t-1} \end{cases}$$

Jeder Zeitschritt wird mit −10 Punkten bestraft, um schnellere Bestückung zu erzielen:

$$r_t = -10$$

Die Überschreitung der Arbeitsraumgrenzen von ±2 mm vom Startpunkt in der horizontalen Ebene führt zur Beendigung der Episode mit einer Bestrafung von −150 Punkten:

$$r_w = -150, falls\ \max\left(|x|, |y|\right) > 2\ mm, sonst\ 0$$

Eine Episode, die zum erfolgreichen Einstecken des Bauteils in die Löcher geführt hat, wird im letzten Zeitschritt mit von 500 Punkten belohnt. Eine Episode wird abgebrochen, falls die Kontaktkraft 7 N übersteigt oder die Dauer der Episode 20 diskrete Zeitschritte erreicht.

Mit dem definierten Beobachtungsraum, Handlungsraum und Belohnungsfunktion liegen alle Voraussetzungen für das Training eines RL-Modells vor. Programmtechnisch wurde die Trainingsumgebung und die anwendungsspezifische Datenvorverarbeitung mithilfe vom Python-Paket „gym“ [24] hinterlegt. Die Lernalgorithmen wurden in der Implementierung aus dem Paket „stable_baselines3“ [25] importiert.

4 Experimentelle Untersuchungen

Die Tauglichkeit des Hexaglide und verschiedener Steuerungsmethoden wird in umfassenden Versuchsreihen überprüft und analysiert. Sowohl die heuristische Methode als auch RL-Modelle greifen auf Bewegungsfähigkeiten des Hexaglide zurück, sodass die benutzte Hardware für beide Steuerungsmethoden in allen Versuchen gleich ist.

4.1 Heuristische Methode

Während der Tests wurde festgestellt, dass die Heuristik einen vielversprechenden Ansatz darstellt, sodass der Umfang der geplanten Experimente entsprechend breit gewählt wurde. Für die Validierung der Methode wurden acht Bauteile an jeweils fünf Zielpositionen auf einer Lochrasterplatte für jeweils zehn Mal getestet (insgesamt 400 Versuche). Die relevanten Bauteildaten sowie die Zielpositionen sind dem **Bild 5** zu entnehmen.

In jedem Versuch wurde die Anzahl der benötigten Zeitschritte und das Endergebnis (Erfolg oder Misserfolg) aufgezeichnet. **Tabelle 1** fasst die Erfolgsquote (Q) und die durchschnittliche Anzahl der Zeitschritte (ØT) pro Bauteil und Zielposition zusammen. Im Durchschnitt dauerte ein Einsteckversuch 9,9 Zeitschritte, insgesamt waren 383 aus 400 Versuchen erfolgreich (Q = 95,8 %). Die Bauteile A, D und G wurden in allen 50 damit durchgeführten Versuchen erfolgreich platziert.

Trotz der hohen Erfolgsquote lassen sich mehrere Fehlerquellen beobachten. Die meisten Fehler wurden dadurch verursacht, dass die optisch nicht kontrollierten, hinteren Pins nicht eingesteckt wurden, obwohl der Referenz-Pin richtig zentriert wurde.

https://doi.org/10.37544/1436-4980-2024-09-67 https://www.inlibra.com/de/page/agb - Open Access - CC BY

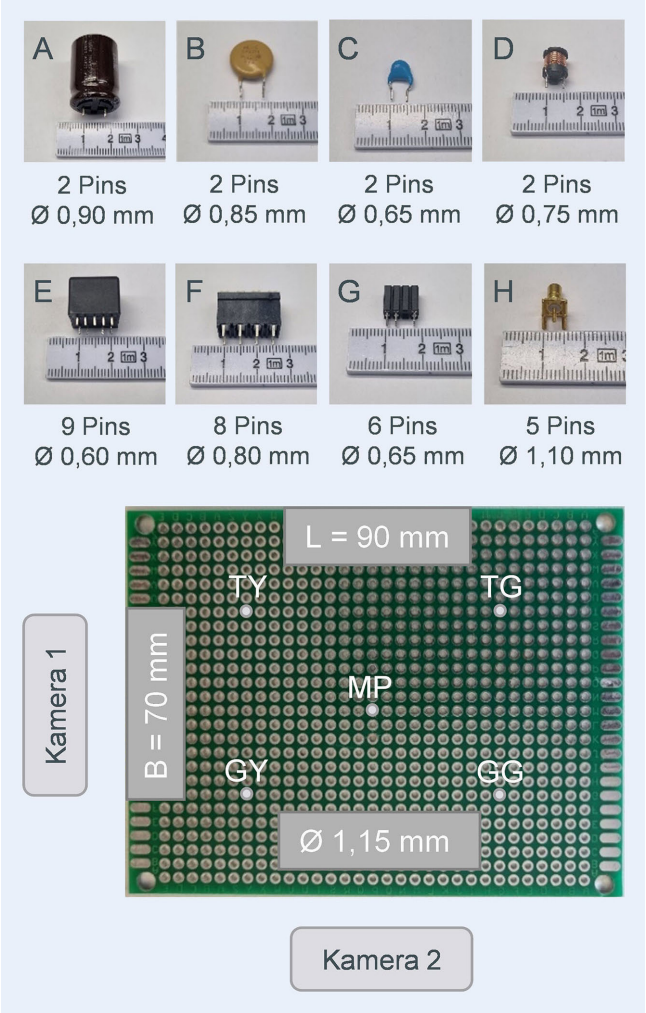

Bild 5. Testbauteile und -leiterplatte für die Validierung der heuristischen Methode. *Foto: IPS*

Das kann dadurch erklärt werden, dass die hier untersuchte Heuristik im Vergleich zur früheren Veröffentlichung [18] keine Rotationsschritte beinhaltet, um Vergleichbarkeit mit den später untersuchten RL-Methoden zu ermöglichen. Die optische Referenzierung kann aufgrund weniger eindeutigen Spitzen von Pins mit einem höheren Durchmesser bis zu circa 2° Winkelfehler aufweisen, was im Suchvorgang berücksichtigt werden soll. Aufgrund von manueller Positionierung der Bauteile im Robotergreifer waren Verkippungen von ebenfalls circa 2° möglich. Sonstige Misserfolge der Heuristik hatten falsche erstmalige Eingaben der Zielpixel in den Bildern oder das Verrutschen der Bauteile im Robotergreifer als Ursache.

Anhand der Daten über die Einsteckdauer (**Bild 6**) lässt sich erkennen, dass die Heuristik tendenziell mehr Zeit in den Positionen braucht, die ungleich weit von den Kameras entfernt sind (TY und GG). Die Bauteilbeschaffenheit hat eine gewisse Auswirkung auf die Zeitschritte, was sich in der größeren Streuung für die Bauteile B, E, F und H wiederspiegelt. Das Bauteil B hat im Gegensatz zum Bauteil A mit ähnlichem Pin-Durchmesser sehr flache, stumpfe Spitzen. Zum einen verbessern schräg abgeschnittene Beinchen die Genauigkeit der optischen Detektion der Pin-Spitzen, zum anderen erhöhen sie die Erfolgswahrscheinlichkeit der Spiralsuche, da die Spitze in mehr Positionen über dem Loch ohne Kontaktkräfte eingefügt werden kann. Die Bauteile E beziehungsweise F haben jeweils 9 beziehungsweise 8 Pins, was die Wahrscheinlichkeit von Kontaktkräften erhöht, wenn mindestens ein Pin leicht verbogen ist. Beim Bauteil H handelt es sich um eine Sonderform mit fünf dicken und stumpfen Beinchen und vereint somit die nachteiligen Eigenschaften der Bauteile B, E und F.

Insgesamt zeigte die heuristische Methode vielversprechende Ergebnisse für Bauteile mit unterschiedlichen Eigenschaften und unterschiedliche Einsteckpositionen.

4.2 Reinforcement Learning

Lernalgorithmen DQN [26] und PPO [27] wurden zum Vergleich für das Training angewendet. DQN wurde mit einer Lernrate von 0,001 und einer Batchgröße von 16 trainiert. PPO wurde mit Lernrate von 0,0002, einer Schrittanzahl von 128 und einem Entropiekoeffizienten von 0,001 optimiert. Die Optimierung der Modellgewichte erfolgte in beiden Fällen mit Adam-Optimierer. Beide Modelle wurden für 10 000 Zeitschritte in der realen Umgebung trainiert, was der Trainingsdauer von zwei Arbeitswochen entspricht, da die Bauteile bei Verformung entweder korrigiert oder ersetzt werden müssen. Als Versuchsbauteil wurde eine Diode mit zwei Pins und 0,2 mm Fügespiel auf der entsprechenden Leiterplatte verwendet. Während des Trainings wurden Fortschrittskurven ermittelt, die die zeitliche Entwicklung der durchschnittlichen Episodenlänge und der durchschnittlichen Belohnung darstellen, **Bild** 7.

Anhand der Trainingsverläufe lässt sich festhalten, dass PPO in der Lage war, im Vergleich zu DQN kürzere Versuche zu erzielen (Bild 7 links), was durch erfolgreiches Einstecken oder durch

Tabelle 1. Versuchsergebnisse mit der Heuristik.

Ziel	Bauteil															
	A		B		C		D		E		F		G		H	
	Q	ØT	Q	ØT	Q	ØT	Q	ØT	Q	ØT	Q	ØT	Q	ØT	Q	ØT
MP	10/10	5,9	10/10	7,7	9/10	12,4	10/10	10,0	8/10	14,6	10/10	8,4	10/10	6,7	9/10	9,3
TG	10/10	6,2	9/10	6,3	9/10	10,4	10/10	15,6	10/10	8,3	10/10	6,8	10/10	5,7	10/10	7,3
TY	10/10	8,0	9/10	9,5	10/10	8,6	10/10	9,3	10/10	7,3	10/10	15,7	10/10	8,7	9/10	15,1
GY	10/10	6,6	8/10	26,1	10/10	5,8	10/10	6,8	9/10	15,2	9/10	10,7	10/10	6,0	10/10	6,6
GG	10/10	13,4	10/10	10,6	9/10	9,3	10/10	8,8	10/10	13,2	7/10	17,0	10/10	6,5	9/10	9,4

https://doi.org/10.37544/1436-4980-2024-09-67 https://www.inlibra.com/de/page/agb - Open Access -

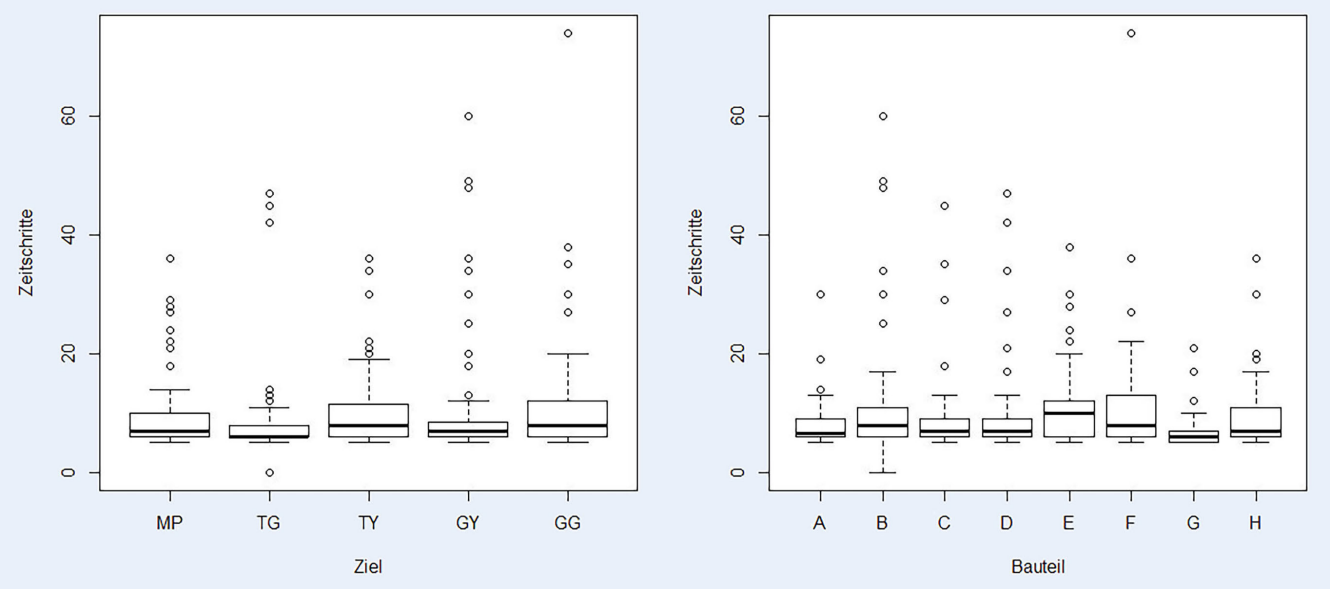

Bild 6. Boxplot-Diagramme der benötigten Zeitschritte in Abhängigkeit vom Ziel (links) und Bauteil (rechts). *Grafik: IPS*

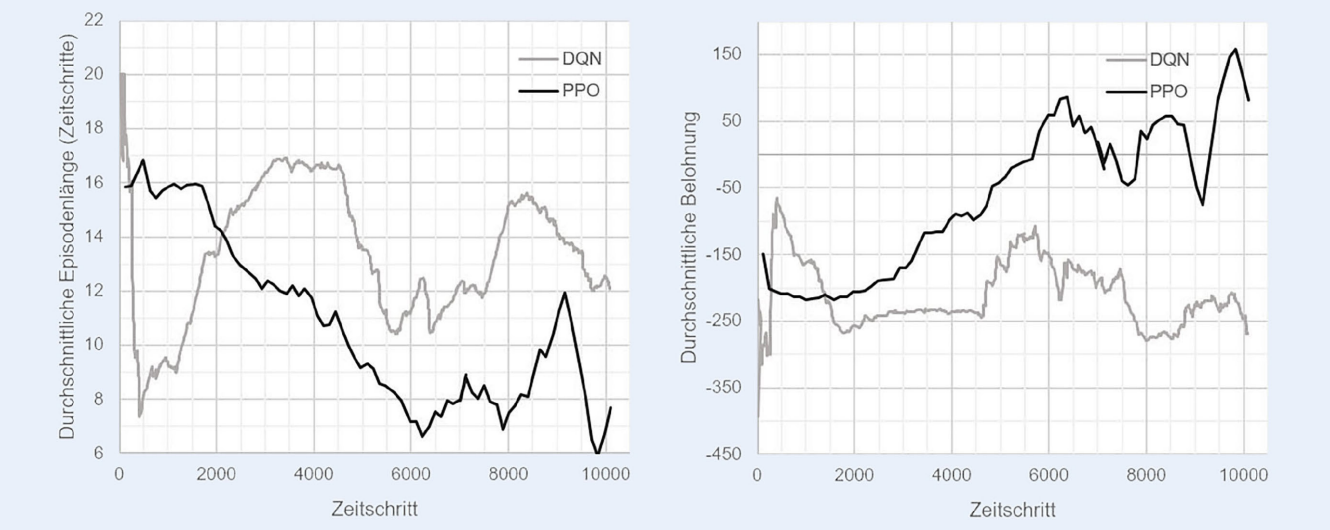

Bild 7. Trainingskurven von DQN und PPO. *Grafik: IPS*

Tabelle 2. Versuchsergebnisse mit RL und Vergleich mit zufälliger Bewegungsstrategie.

Algorithmus	Anzahl Episoden	Anzahl Erfolge	Erfolgsquote	Verhältnis zum Zufall
PPO	1062	176	16,6%	436,8
DQN	773	45	5,8%	152,6

sehr schnelles Erreichen von maximal zulässigen Kontaktkräften möglich ist. PPO war bereits nach 2000 Schritten im Durchschnitt schneller als DQN und hat in den letzten 4000 Zeitschritten mit Fluktuationen ca. acht Schritte pro Episode gebraucht, während DQN über zwölf Schritte benötigte. Die durchschnittliche Belohnung (Bild 7 rechts) war bei PPO nach 1500 Schritten konsistent besser als bei DQN. Abgesehen von großen Schwankungen konnte PPO nach 6000 Zeitschritten positive summierte Belohnungen pro Episode bekommen und erzielte im Maximum eine laufende durchschnittliche Belohnung von 150 Punkten.

Obwohl PPO im Vergleich zu DQN positivere Ergebnisse erzielt, sind die gesamten Erfolgsquoten der beiden Algorithmen niedrig (**Tabelle 2**). Im Vergleich zu einer rein zufälligen Bewegungsstrategie sind diese Ergebnisse nichtsdestotrotz mehr als 100 Mal besser. Dies wurde in einer vereinfachten geometrischen Simulation ermittelt, indem der Bewegungsraum und die Leiterplatte entsprechend der tatsächlich ausgeführten Schrittweite von 0,5 mm diskretisiert wurde. In 1 Millionen stochastischen Simulationsabläufen wurde jede Bewegung zufällig aus den sechs verfügbaren Aktion ausgewählt, mit gleichen Terminierungsbedingungen als für PPO und DQN. Der Anteil der erfolgreichen Be-

https://doi.org/10.37544/1436-4980-2024-09-67 https://www.inlibra.com/de/page/agb - Open Access - CC BY

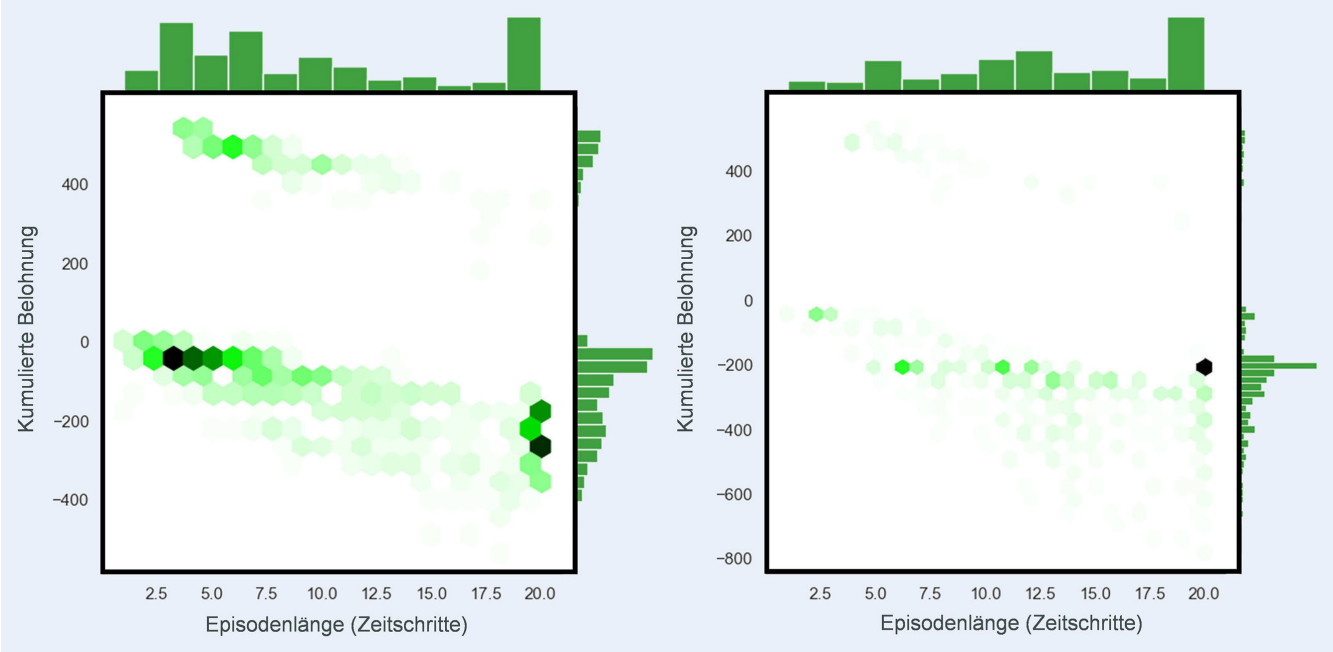

Bild 8. Histogramme zur Analyse von RL-Trainingsepisoden. PPO (links) und DQN (rechts). *Grafik: IPS*

wegungstrajektorien konvergierte gegen 0,038 %, sodass die RL-Ergebnisse in Verhältnis zu dieser Größe gesetzt werden können.

Zur Interpretation des Modellverhaltens kann ein zweifaktorielles Histogramm benutzt werden, das die Verteilung von Episoden anhand ihrer Dauer und kumulierter Belohnungen abbildet , **Bild 8**. Am Histogramm für das PPO-Training wird deutlich, dass es drei wesentliche Cluster von Episoden gibt. In vielen Episoden wird die maximale Anzahl von Zeitschritten (20) erreicht, was bei einer Bestrafung jedes Zeitschritts mit −10 am Episodenende einer kumulierten Belohnung von circa −200 entspricht, falls keine hohen Kontaktkräfte ausgeübt wurden und das Bauteil überhaupt in Richtung der Leiterplatte bewegt wurde. Das zweite Cluster hat die höchste Konzentration bei den Episoden mit einer Dauer von 3–4 Zeitschritten und Belohnung von etwa −50. Diese Gruppe stellt Versuche dar, bei denen das Bauteil direkt gegen die Leiterplatte gefahren wurde, sodass aufgrund von hohen Kräften schnell abgebrochen werden musste. Der „Schweif“ dieses Clusters repräsentiert Episoden, die entweder wegen dem Überschreiten der sinnvollen Arbeitsraumgrenzen oder wegen der hohen Kontaktkraft scheiterten. Klar getrennt von den ersten beiden Clustern sind Episoden mit mehr als 400 Punkten kumulierter Belohnung. Dieser hohe Wert kann dank der Belohnungsfunktion nur aufgrund von einer finalen Erfolgsbelohnung von 500 vorkommen. Vom Grundsatz her sind die Cluster von DQN-Episoden ähnlich, allerdings lässt sich hier eine extrem hohe Konzentration bei Episodendauern von 20 Schritten und −200 Belohnung feststellen. Das korreliert mit subjektiver Beobachtung während des Trainings, dass DQN vorwiegend zufällige Bewegungen ausgeführt und Berührungen mit der Leiterplatte vermieden hat.

Die Untersuchung von RL hat zu keinen befriedigenden Ergebnissen hinsichtlich praktischer Anwendbarkeit geführt, da das Training sehr zeitintensiv ist und kein zuverlässiges Steuerungsmodell liefert. Von der Übertragbarkeit des Modells auf andere Bauteile kann nach aktuellem Stand nicht ausgegangen werden.

5 Diskussion und Ausblick

Um die Ressourceneffizienz der Elektronikfertiger im Bereich der THT-Bestückung zu steigern, müssen technische Lösungen entwickelt werden, die gezielt die erforderliche Leistung anbieten. Dies kann durch die auf den Anwendungsfall zugeschnittene Sensorik und Hardware erfolgen, sodass die technologischen Investitionen optimal genutzt werden. Am Beispiel des vorgestellten Hexaglide-Manipulators wird verdeutlicht, dass marktübliche Kameratechnik eine ausreichende Auflösung besitzt, um THT-Bauteile erfolgreich zu detektieren und zur Leiterplatte zu referenzieren. Darüber hinaus zeigten die experimentellen Untersuchungen, dass die senkrechte Komponente der Kontaktkraft als eines der Signale für die Steuerung des Einsteckvorgangs genügt und mit kostengünstigen Sensoren erfasst werden kann.

Im diskutierten Anwendungsfall der Leiterplattenbestückung versetzen robuste Bildverarbeitungsalgorithmen den Hexaglide in die Lage, mit Bauteilen in geringeren Ordnungszuständen umzugehen, wodurch die Anforderungen an die Bauteilbereitstellung sinken. Im aktuellen Entwicklungsstand der Bildverarbeitung sollen die Bauteile im Greifer lediglich nicht verkippt sein; mechanisch ist der Hexaglide aber bereits in der Lage, die Positionierplattform zu neigen. Zur Steuerung der unmittelbaren Einsteckbewegung können verschiedene Methoden herangezogen werden – in diesem Beitrag wurde eine heuristische Methode und zwei RL-Methoden untersucht. Die Heuristik erzielt 95,8 % Erfolgsquote, während RL nach 10 000 Trainingszeitschritten mit 16,6 % deutlich schlechter abschneidet. Allerdings ist das bessere RL-Modell PPO mehr als 400-fach erfolgreicher als eine Strategie, die jeden Bewegungsschritt zufällig auswählt. Das deutet auf Vorhandensein gewisser Lernfähigkeit des Algorithmus. Die Trainingsdauer und der manuelle Betreuungsaufwand der RL-Modelle kann reduziert werden, indem Bauteile zwischen Episoden automatisch ausgetauscht werden. Dafür muss das System mit flexibler und zuverlässiger Zuführungstechnik für THT-Bauteile ausgestattet sein, sodass jeglicher menschliche Eingriff während

https://doi.org/10.37544/1436-4980-2024-09-67 https://www.inlibra.com/de/page/agb - Open Access - CC BY

des Trainings entfällt. Alternativ kann RL in Simulation trainiert werden. Um die Übertragbarkeit in die Realität zu gewährleisten, muss die Simulation physikalische Wechselwirkungen zwischen Leiterplatte, Beleuchtung, Bauteil und Robotergreifer realitätstreu abbilden.

Eine hervorzuhebende Eigenschaft und Unterschied der RL-Methode von der heuristischen Methode ist die Notwendigkeit eines Trainings, um das Modell an die Aufgabe anzupassen. Zusätzlich sind RL-Algorithmen nicht komplett frei von Parametern, deren Einstellung das Lernergebnis stark beeinflussen kann und viel Aufwand für die Bewertung benötigt. Die Bedeutung der Hyperparameter für RL (zum Beispiel Batchgröße, Lernrate, Explorationsrate) ist für einen industriellen Anwender in der Regel nicht direkt bekannt und die optimale Parameterempfehlung kann je nach Aufgabenkomplexität stark variieren. Die Parameter der heuristischen Methode wie die Schrittweite, die Reihenfolge der Suchbewegungen oder die Pixeltoleranz der Pin-Zentrierung haben direkte und nachvollziehbare Effekte auf das Einstecken und können somit kurzzyklisch durch Anlagenbetreuende angepasst und optimiert werden.

Aus Hardwaresicht besitzt ein Hexaglide als geschlossene kinematische Kette eine hohe Steifigkeit und Genauigkeit. Diese unterscheiden diesen Manipulator von vielen marktüblichen Knickarmrobotern. Obwohl für das Greifen und Bringen der Bauteile zum Hexaglide ein anderer Roboter benötigt wird, muss dieser keine hohe Genauigkeit aufweisen, weil der Hexaglide die hohe Bewegungspräzision mitbringt. Die Integration von Kameras und Kraftsensoren mit dem Leichtbauroboter ist nicht mehr erforderlich. Lediglich der Austausch von einfachen Statusmeldungen zur Kommunikation zwischen zwei Robotern muss vorhanden sein.

Neben der THT-Bestückung können Hexaglides als universelle Manipulatoren mit sechs Freiheitsgraden für andere Zwecke eingesetzt werden. Besonders vorteilhaft bei dieser Kinematik sind hohe Steifigkeit, geringe zu bewegende Massen und Robustheit gegen Unstetigkeit und Sprüngen in den Gelenkstellungen.

Zukünftig sind weitere Verbesserungen in der Auslegung des Hexaglides und der Steuerungsalgorithmen denkbar. Darüber hinaus besteht Entwicklungsbedarf an einfachen Vereinzelungs- und Bereitstellungsvorrichtungen für THT-Bauteile, um die Notwendigkeit eines separaten Leichtbauroboters in dem Montagesystem zu eliminieren.

FÖRDERHINWEIS

Das Projekt „SmartAssembly“ (Förderkennzeichen: KK5072203WO0) wird vom Bundesministerium für Wirtschaft und Klimaschutz (BMWK) aufgrund eines Beschlusses des Deutschen Bundestages gefördert.

Literatur

[1] Brindley, K.: Newnes Electronics Assembly Handbook. Oxford: Heinemann Professional Publishing Ltd 1990
[2] Risse, A.: Fertigungsverfahren der Mechatronik, Feinwerk- und Präzisionsgerätetechnik. Wiesbaden: Vieweg+Teubner Verlag 2012
[3] Feldmann, K.: Montage in der Leistungselektronik für globale Märkte. Berlin, Heidelberg: Springer Berlin Heidelberg 2009
[4] Lotter, B.; Wiendahl, H.-P.: Montage in der industriellen Produktion. Ein Handbuch für die Praxis. Berlin, Heidelberg: Springer Vieweg 2012
[5] Metzner, M.; Reisinger, D.; Ortmann, J.-N. et al.: An Approach for Direct Offline Programming of High Precision Assembly Tasks on 3D Scans Using Tactile Control and Automatic Program Adaption. In: Schüppstuhl, T.; Tracht, K.; Raatz, A. (Hrsg.): Annals of Scientific Society for Assembly, Handling and Industrial Robotics 2021. Cham: Springer International Publishing 2022, S. 215–225
[6] Domm, M.: Kleinserienbestückung von Leiterplatten mit bedrahteten Bauelementen durch Industrieroboter. Dissertation, Universität Stuttgart, 1990
[7] Ma, Y.; Xie, Y.; Zhu, W. et al.: An Efficient Robot Precision Assembly Skill Learning Framework Based on Several Demonstrations. IEEE Transactions on Automation Science and Engineering 20 (2023) 1, S. 124–136
[8] Haugaard, R. L.; Buch, A. G.; Iversen, T. M.: Self-supervised deep visual servoing for high precision peg-in-hole insertion, 2022
[9] Alt, B.; Katic, D.; Jäkel, R. et al.: Heuristic-free Optimization of Force-Controlled Robot Search Strategies in Stochastic Environments 18 (2022), S. 8887–8893
[10] Bartyzel, G. M.; Półchłopek, W.; Rzepka, D.: Vision-driven Deep Reinforcement Learning for Electronic Components Robotic Insertion Tasks. 2022
[11] Metzner, M.; Leurer, S.; Handwerker, A. et al.: High-precision assembly of electronic devices with lightweight robots through sensor-guided insertion. Procedia CIRP 97 (2021), S. 337–341
[12] pi4_robotics: THT Placer. Internet: https://www.pi4.de/product-thtp1.html. Zugriff am 10.08.2023
[13] ABB: YuMi takes over THT assembly. Internet: https://new.abb.com/news/detail/73577/cstmr-yumi-takes-over-tht-assembly. Zugriff am 10.08.2023
[14] ArtiMinds Robotics GmbH: THT Assembly with Robots Boosts Automation instead of Manual Labor. Internet: https://www.ien.eu/article/tht-assembly-with-robots-boosts-automation-instead-of-manual-labor/. Zugriff am 10.08.2023
[15] Hesse, S.: Grundlagen der Handhabungstechnik. München: Hanser Verlag 2020
[16] Honegger, M.; Codourey, A.; Burdet, E.: Adaptive control of the Hexaglide, a 6 dof parallel manipulator. International Conference on Robotics and Automation, Albuquerque, NM, USA, 1997, S. 543–548
[17] Palpacelli, M.-C.; Carbonari, L.; Palmieri, G. et al.: Functional Design of a 6-DOF Platform for Micro-Positioning. Robotics 9 (2020) 4, S. 99
[18] Polikarpov, M.; Mehmood, Y.; Boiar, D. et al.: A Cost-Efficient Robotic Through-Hole Assembly System for Dual-Pin Component Insertion. 2023 11th International Conference on Control, Mechatronics and Automation (ICCMA), Grimstad, Norway, 2023, S. 271–277
[19] Dombrowski, U.; Stefanak, T.; Mönning, T.: Modulares MRK-Bausteinsystem. Zeitschrift für wirtschaftlichen Fabrikbetrieb 113 (2018) 10, S. 627–632
[20] Sutton, R. S.; Bach, F.; Barto, A. G.: Reinforcement Learning. An Introduction. Massachusetts: MIT Press Ltd 2018
[21] Hosna, A.; Merry, E.; Gyalmo, J. et al.: Transfer learning: a friendly introduction. Journal of big data 9 (2022) 1, p. 102
[22] Deng, J.; Dong, W.; Socher, R. et al.: ImageNet: A large-scale hierarchical image database. 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops), Miami, FL, 62009, S. 248–255
[23] He, K.; Zhang, X.; Ren, S. et al.: Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 62016, S. 770–778
[24] Brockman, G.; Cheung, V.; Pettersson, L. et al.: OpenAI Gym, 2016
[25] Raffin, A.; Hill, A.; Gleave, A. et al.: Stable-Baselines3: Reliable Reinforcement Learning Implementations. Journal of Machine Learning Research 22 (2021), S. 1–8
[26] Mnih, V.; Kavukcuoglu, K.; Silver, D. et al.: Human-level control through deep reinforcement learning. Nature 518 (2015) 7540, pp. 529–533
[27] Schulman, J.; Wolski, F.; Dhariwal, P. et al.: Proximal Policy Optimization Algorithms, 2017. Internet: https://arxiv.org/abs/1707.06347. Zugriff am 13.05.2024

https://doi.org/10.37544/1436-4980-2024-09-67 https://www.inlibra.com/de/page/agb - Open Access - (cc) BY

Mikhail Polikarpov, M. Sc.
mikhail.polikarpov@tu-dortmund.de
Tel. +49 231 / 755-5767
Foto: Autor

Yousuf Mehmood, M. Sc.

Daniel Boiar, M. Sc.

Univ.-Prof. Dr.-Ing. Jochen Deuse

Technische Univerität Dortmund
Institut für Produktionssysteme
Leonhard-Euler-Str. 5, 44227 Dortmund
ips.mb.tu-dortmund.de

LIZENZ

Dieser Fachaufsatz steht unter der Lizenz Creative Commons Namensnennung 4.0 International (CC BY 4.0)

https://doi.org/10.37544/1436-4980-2024-09-67 https://www.inlibra.com/de/page/agb - Open Access -