

GEFÖRDERT VOM



Bundesministerium
für Bildung
und Forschung



DLR Projektträger

Abschlussbericht – Teil 2: Ausführlicher Bericht

Berichtszeitraum: 1.07.2020 – 31.12.2022

FKZ 01QE1927A

Laufzeit des Vorhabens: 01.07.2020 – 31.12.2022

Vorhabenbezeichnung

Autonome visuelle, industrielle Roboterinteraktion durch Künstliche Intelligenz (AutoVikki)

Teilvorhaben

KI-Vision-System für industrielle, autonome und robotische Einzelfertigung

Autoren

Patrick Selle

Zuwendungsempfänger:

Hahn-Schickard-Gesellschaft für angewandte Forschung e.V. (HS)

Wilhelm-Schickard-Straße 10

78052 Villingen-Schwenningen

Deutschland

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS20039B gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt bei der Autorin/Autor.

Vorsitzender des Vorstands:
Prof. Dr. Volker Nestle

Geschäftsführer:
Clemens Pecha

Institutsleiter:
Prof. Dr. Oliver Amft
Prof. Dr.-Ing. Alfons Dehé
Dr.-Ing. Karl-Peter Fritz
Prof. Dr. techn. Boris Mizaikoff
Apl. Prof. Dr. Felix von Stetten
Prof. Dr.-Ing. Roland Zengerle
Prof. Dr.-Ing. André Zimmermann

Sparkasse Schwarzwald-Baar
BIC: SOLADES1VSS
IBAN: DE82 6945 0065 0000 0927 26
USt-IdNr.: DE147807640

Seite 1 von 19

1. Darstellung der durchgeführten Arbeiten im Vergleich zur Vorhabenbeschreibung, die wichtigsten Positionen des zahlenmäßigen Nachweises sowie die Notwendigkeit und Angemessenheit der geleisteten Projektarbeiten

AP1 - Projektmanagement

Im Rahmen des Projekts wurde ein effektives Projektmanagement durchgeführt, das eine erfolgreiche Zusammenarbeit der Partner im Konsortium ermöglicht hat. Trotz des Personalwechsels projektrelevanter Personen verlief die Zusammenarbeit gut, eng und konstruktiv. Hierbei fand ein enger inhaltlicher Austausch auf verschiedenen Ebenen statt, der durch regelmäßige Videokonferenzen und die Nutzung einer gemeinsamen GitLab-Plattform unterstützt wurde.

Aufgrund der Corona-Pandemie fand seit dem Jahr 2020 nur ein physisches Treffen aller Projektteilnehmer bei OFK statt. Zusätzlich ergab sich auf Grund der räumlichen Nähe einer anderen Dienstreise die Möglichkeit, dass am eigentlich virtuellen Kickoff-Meeting am 6. Juli 2020 Prof. Sikora physisch bei BBS teilnehmen konnte. Für alle anderen Treffen wurden virtuelle Formen genutzt, um die Zusammenarbeit voranzutreiben.

1. Projekttreffen	1. Otto Fuchs KG	2. Meinerzhagen	3. 7. Oktober 2022
-------------------	------------------	-----------------	--------------------

Hahn-Schickard war während des gesamten Projektverlaufs aktiv in die Projektleitung eingebunden, wobei der Hahn-Schickard-Projektleiter Patrick Selle sowie die Projektmitarbeiter Maximilian Schwäglar und später Daniel Hauser an allen Webkonferenzen teilnahmen.

Insgesamt konnte das Projektmanagement dazu beitragen, den Projektfortschritt zu überwachen und sicherzustellen, dass das Projektziel im Wesentlichen erfolgreich erreicht werden konnte.

AP2 - Anwendungsszenarien & Systemanforderungen

Im Rahmen des AutoVikki Projekts wurden zwei Anwendungsszenarien identifiziert, die automatisierte In-Vitro-Pflanzenvermehrung und die Nachbearbeitung geschmiedeter Bauteile. Auf Grund der mangelnden Zuarbeit der BBS konzentrierten sich die Entwicklungsarbeiten insbesondere auf das Szenario B, für das zwei Bauteile identifiziert wurden: Werkzeug-Nr. 36851 mit einem aus dem Schmiedeprozess zurückgebliebenen Grat und Werkzeug-Nr. 33000 mit einem aus dem Schmiedeprozess zurückgebliebenen Nocken, gezeigt in Abbildung 1.

Das Ziel des Projekts ist es, die Bauteile automatisch aus einer mechanischen Rutsche zu greifen, vor die 3D-Kamera zu heben und eine Punktwolke des Objekts zu erzeugen. Anschließend sollen verschiedene maschinelle Lernmethoden untersucht, miteinander verglichen und die effizienteste eingesetzt werden, um die Fehler und ihre Ausprägungen zu erkennen und dem Schleifprozess zu übergeben.

Für die Umsetzung dieses Ziels wurden spezifische Systemanforderungen identifiziert. Hierzu zählen insbesondere die Notwendigkeit einer präzisen mechanischen Vorrichtung zur Greifung und Handhabung der Bauteile, eine hohe Auflösung und Genauigkeit der 3D-Kamera sowie die Fähigkeit, große Mengen an Daten zu verarbeiten und zu analysieren.

Darüber hinaus wurden spezifische Anforderungen für die Anwendungsszenarien identifiziert. Für die Nachbearbeitung geschmiedeter Bauteile ist eine hohe Genauigkeit und Reproduzierbarkeit des Schleifprozesses sowie eine effiziente Integration in bestehende Produktionsprozesse erforderlich.

Vorsitzender des Vorstands:
Prof. Dr. Volker Nestle

Geschäftsführer:
Clemens Pecha

Institutsleiter:
Prof. Dr. Oliver Amft
Prof. Dr.-Ing. Alfons Dehé
Dr.-Ing. Karl-Peter Fritz
Prof. Dr. techn. Boris Mizaikoff
Apl. Prof. Dr. Felix von Stetten
Prof. Dr.-Ing. Roland Zengerle
Prof. Dr.-Ing. André Zimmermann

Sparkasse Schwarzwald-Baar
BIC: SOLADES1VSS
IBAN: DE82 6945 0065 0000 0927 26
USt-IdNr.: DE147807640

Seite 2 von 19

Zusammenfassend kann festgestellt werden, dass im Rahmen des AutoVikki Projekts spezifische Anwendungsszenarien und Systemanforderungen identifiziert wurden, um das Ziel der automatisierten Fehlererkennung und -behebung zu erreichen. Die Umsetzung erfordert präzise mechanische Vorrichtungen, hochauflösende 3D-Kameras, effiziente Datenverarbeitung und -analyse sowie spezifische Anforderungen für die einzelnen Anwendungsszenarien.



Abbildung 1 Links: zeigt das Werkzeugteil 036851 mit dem aus dem Schmiedeprozess zurück gebliebenen Grat. Rechts: zeigt das Werkzeugteil 33000 mit dem aus dem Schmiedeprozess zurück gebliebenen Nocken.

AP3: Systemkonzept und Architektur Aspekte der Roboterzelle

AP3.1: Konzeptionierung der Hardware: Sensorik und Robotik

HS war nicht Teil dieses Teilarbeitspaketes.

AP3.2: Konzeptionierung der zentralen, herstellerunabhängigen Kommunikationsplattform

HS war nicht Teil dieses Teilarbeitspaketes.

AP3.3: Konzeptionierung einer herstellerunabhängigen Schnittstellenebene

HS war nicht Teil dieses Teilarbeitspaketes.

AP3.4: Konzeptionierung einer Datenbank und Cloud Umgebung

Das Federated Learning Framework Flower¹ ist darauf ausgelegt, maschinelles Lernen auf verteilten Geräten durchzuführen, die untereinander kommunizieren und Daten austauschen können. Eine Cloud-Umgebung, die für den Einsatz von Flower geeignet ist, sollte daher einige Anforderungen erfüllen, wie z.B.:

¹ <https://github.com/adap/flower>

- **Unterstützung von gRPC:** Flower verwendet gRPC für die Kommunikation zwischen den verteilten Geräten. Daher sollte die Cloud-Umgebung gRPC unterstützen, um eine nahtlose Integration mit Flower zu ermöglichen.
- **Skalierbarkeit:** Da Flower für verteiltes maschinelles Lernen auf einer großen Anzahl von Geräten ausgelegt ist, sollte die Cloud-Umgebung in der Lage sein, die Skalierung von Ressourcen zu unterstützen, um eine ausreichende Kapazität für die Verarbeitung von Daten und Modellen bereitzustellen.
- **Sicherheit:** Da Flower mit Daten arbeitet, die auf verschiedenen Geräten verteilt sind, muss die Cloud-Umgebung Sicherheitsmechanismen bereitstellen, um Datenverlust, Datenschutzverletzungen oder unbefugte Zugriffe zu vermeiden.
- **Hochverfügbarkeit:** Eine Cloud-Umgebung für Flower sollte eine hohe Verfügbarkeit gewährleisten, um sicherzustellen, dass der Lernprozess auf den verteilten Geräten stets stabil und ununterbrochen durchgeführt werden kann.
- **Leistung:** Da Flower für maschinelles Lernen auf verteilten Geräten konzipiert ist, ist eine hohe Leistungsfähigkeit der Cloud-Umgebung für eine schnelle Verarbeitung von Daten und Modellen erforderlich.

AP3.5: Bedienkonzept

HS war nicht Teil dieses Teilarbeitspaketes.

AP4: Konzeptionierung, Implementierung, Training und Validierung der KI

AP4.1: Sammeln von Trainingsdaten in den beiden Szenarien

In diesem Teilarbeitspaket waren die Projektpartner HS, BBS, AIT und OFK unter Koordination von HS aktiv.

Im Rahmen des Forschungsprojektes AutoVikki wurde in Szenario B der Schwerpunkt auf die Erkennung von Graten und Nocken auf zwei Werkzeugteilen mit den Werkzeugnummern 36851 und 33000 gelegt. Ziel des Arbeitspaketes 4.1 war es, eine umfangreiche Sammlung von Trainingsdaten für diese Erkennung zu sammeln und zu labeln.

Als Eingabe für die Erkennung der Grate und Nocken wurden 3D-Punktwolken von der 3D-Kamera Zivid One Plus S verwendet. Diese Daten wurden mit einer 2D-Segmentierungsmaske gelabelt, die mit einem speziellen Labelframework erstellt wurde. Das Labelframework wurde bei Hahn-Schickard entwickelt und Otto Fuchs KG bereitgestellt und in dessen Verwendung unterwiesen.

Abbildung 2 und 3 zeigen die Vorrichtungen, Eingaben und Ergebnisse der Entwicklung des Labelframeworks beider Werkzeugteile. Das linke Bild zeigt jeweils den provisorischen Messstand, das mittlere Bild die Tiefenbildinformation extrahiert aus der 3D Punktwolke und das rechte Bild das jeweilige Label generiert aus dem Framework.

Vorsitzender des Vorstands:
Prof. Dr. Volker Nestle

Geschäftsführer:
Clemens Pecha

Institutsleiter:
Prof. Dr. Oliver Amft
Prof. Dr.-Ing. Alfons Dehé
Dr.-Ing. Karl-Peter Fritz
Prof. Dr. techn. Boris Mizaikoff
Apl. Prof. Dr. Felix von Stetten
Prof. Dr.-Ing. Roland Zengerle
Prof. Dr.-Ing. André Zimmermann

Sparkasse Schwarzwald-Baar
BIC: SOLADES1VSS
IBAN: DE82 6945 0065 0000 0927 26
UST-IdNr.: DE147807640

Seite 4 von 19

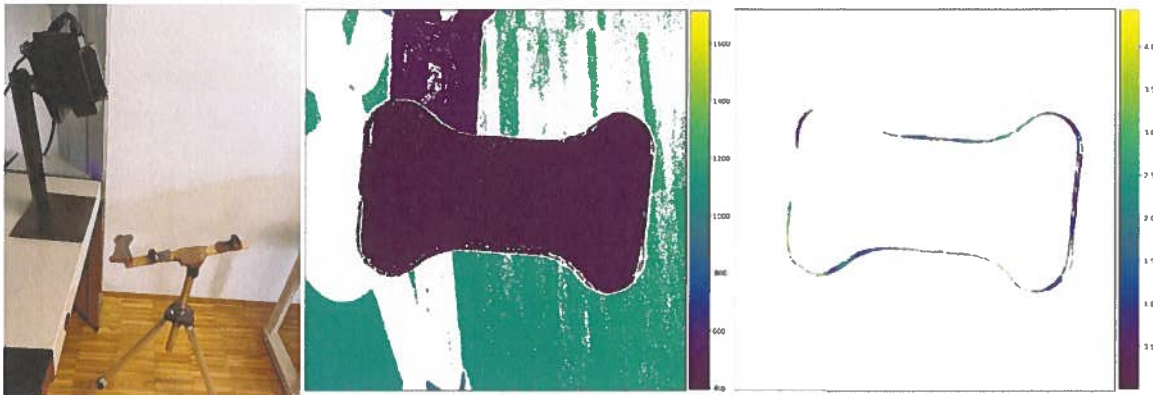


Abbildung 2: Links: Aufnahmevorrichtung der 3D Kamera Zivid One Plus sowie das Werkzeugteil 36851. Mitte: die z-Information als Tiefenbild zeigt den Abstand von Kamera zum Objekt. Die Colorbar gibt diesen in mm an. Rechts: der Grat als Tiefenbildkarte als Ergebnis

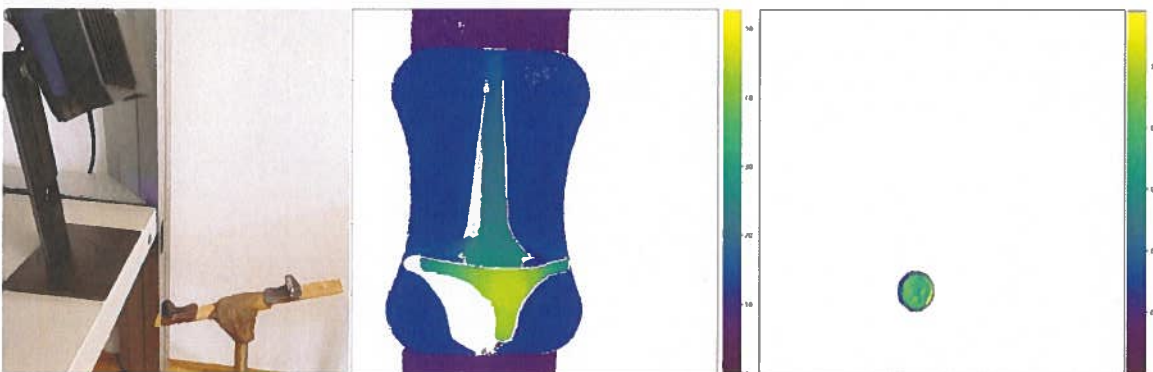


Abbildung 3: Links: Aufnahmevorrichtung der 3D Kamera Zivid One Plus sowie das Werkzeugteil 33000. Mitte: die z-Information als Tiefenbild zeigt den Abstand von Kamera zum Objekt. Die Colorbar gibt diesen in mm an. Rechts: der Nocken als Tiefenbildkarte als Ergebnis

Da jedoch eine Begrenzung in Bezug auf die Verfügbarkeit von Trainingsdaten bestand, wurde das Unternehmen Arnold-IT beauftragt, synthetische Trainingsdaten zu erzeugen. Auf Basis der jeweiligen CAD-Modelle konnten synthetische 3D-Trainingsdaten generiert werden, die den realen Daten ähnlich waren. Nocken und Grate beider Werkzeugteile wurden realitätsnah zufällig generiert. Um die Trainingsdaten weiter zu vervollständigen und ihre Übertragbarkeit auf den Realfall zu verbessern, wurden verschiedene Augmentierungstechniken angewendet. Diese Techniken beinhalten beispielsweise Rotationen, Translationen und Skalierungen von Datenpunkten sowie Rauschreduzierung und Punktwolkenfilterung.

- **Rotation:** Hierbei wird die Punktwolke um eine oder mehrere Achsen gedreht, um neue Ansichten des Objekts zu erzeugen. Die Rotation ist um alle drei verschiedene Achsen und einen zufälligen Winkel erfolgt. Die Rotation einer Punktwolke teilt sich in zwei verschiedene Rotationen auf. 1. Rotation der Punktwolken-Daten: Hierbei werden die Koordinaten der Punktwolkenpunkte rotiert, um neue Ansichten des Objekts aus verschiedenen Blickwinkeln zu erzeugen. Diese Art der Rotation ähnelt der Rotation von 3D-Modellen oder Objekten. 2.

Perspektivische Rotation: Diese Art der Rotation wird verwendet, um Perspektivverzerrungen in der 3D-Punktwolke zu erzeugen, ähnlich wie bei der Rotation von 2D-Bildern.

- **Translation:** Bei der Translation wird die Punktwolke in einer zufälligen Richtung verschoben, um die Position des Objekts im Raum zu verändern. Die Größe der Verschiebung in alle 3 Dimensionen wurde zufällig gewählt.
- **Skalierung:** Hierbei wird die Größe der Punktwolke verändert, um verschiedene Größenverhältnisse des Objekts zu simulieren. Der Skalierungsparameter wurde zufällig gewählt.
- **Rauschen:** Rauschen wurde verwendet, um zufällige Störungen und Abweichungen in der Punktwolke zu erzeugen. Es gibt verschiedene Arten von Rauschen, einschließlich zufälliger Abweichungen von Punktpositionen durch Hinzufügen von zufälligen Punkten. Insbesondere wurden durch diese Technik angewendet, um die Nocken und den Grat variiert.

Ein weiteres angewandtes Rauschverfahren war die Einbringung von Fehlstellen (NaNs), um die synthetischen Daten näher an reale Aufnahmen heranzuführen. Abbildung 4 verdeutlicht diesen Ansatz anhand von zwei Beispielen, bei denen weiße Stellen aufgrund der reflektierenden Beschaffenheit der Werkzeugeiloberfläche nicht als verwendbare 3D-Information in der Punktwolke genutzt werden konnten. Zur Umsetzung dieses Verfahrens wurde ein Zufallsalgorithmus entwickelt, der bestimmte Bereiche in der Punktwolke der synthetisch generierten Trainingsdaten mit NaN-Werten ersetzt. Ziel war es, die Lücke zwischen realer und synthetischer Welt zu schließen und damit die Qualität der synthetischen Daten zu verbessern. Abbildung 5 präsentiert verschiedene Ergebnisse dieses Algorithmus und zeigt die positiven Auswirkungen auf die Qualität der synthetischen Trainingsdaten.

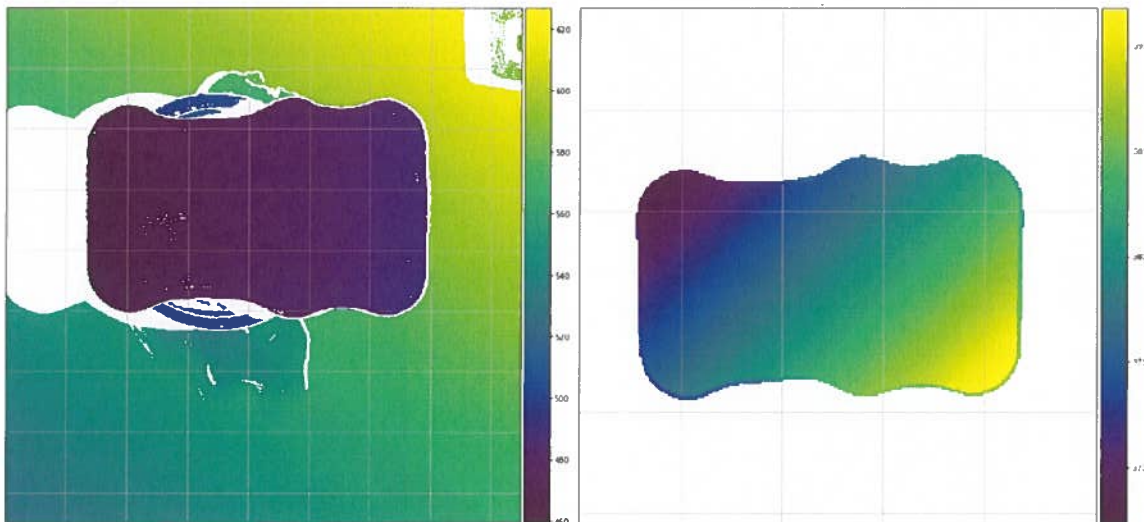


Abbildung 4: Links: z-Information einer realen 3D-Punktwolke mit Hintergrund (Werkzeugteil 36851). Auffällig sind die weißen Stellen auf der Grundplatte des Werkzeugteils in denen keine 3D-Information vorliegt aufgrund von der reflektierenden Oberflächenbeschaffenheit. Rechts: z-Information einer synthetischen baugleichen 3D-Punktwolke ohne Hintergrund. Der hohe Idealisierungsgrad zeigt keinerlei Fehlstellen (weiße Stellen).

Vorsitzender des Vorstands:
Prof. Dr. Volker Nestle

Geschäftsführer:
Clemens Pecha

Institutsleiter:
Prof. Dr. Oliver Amft
Prof. Dr.-Ing. Alfons Dehé
Dr.-Ing. Karl-Peter Fritz
Prof. Dr. techn. Boris Mizaikoff
Apl. Prof. Dr. Felix von Stetten
Prof. Dr.-Ing. Roland Zengerle
Prof. Dr.-Ing. André Zimmermann

Sparkasse Schwarzwald-Baar
BIC: SOLADES1VSS
IBAN: DE82 6945 0065 0000 0927 26
UST-IdNr.: DE147807640

Seite 6 von 19

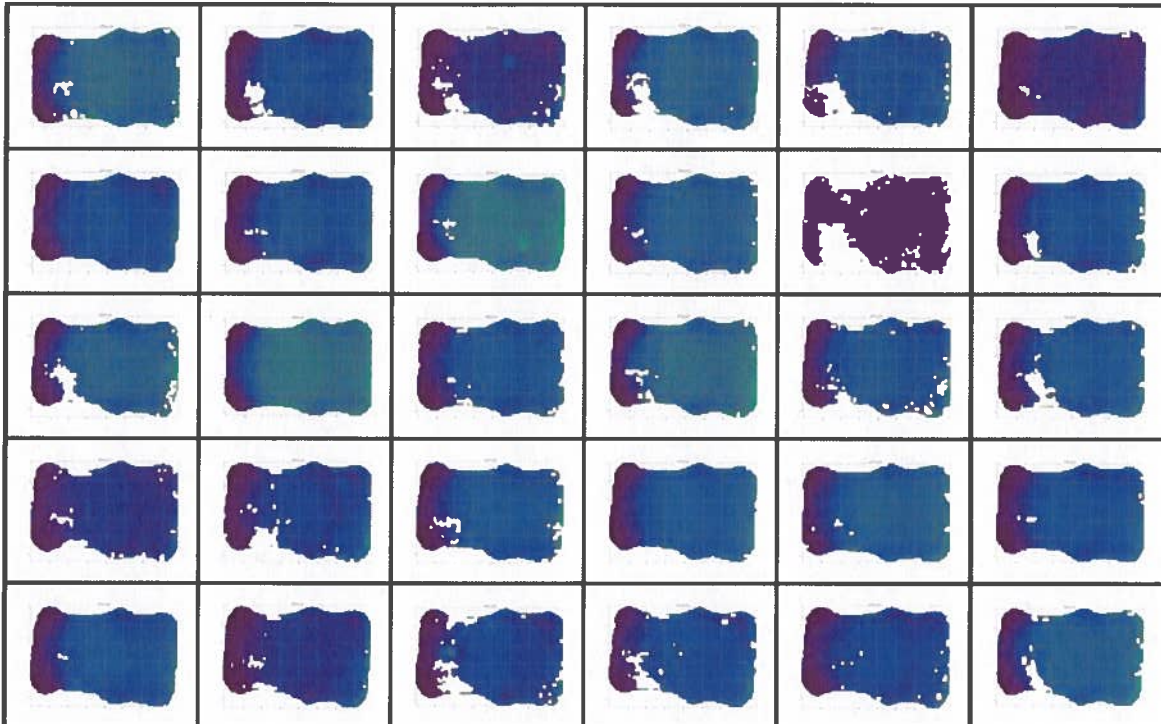


Abbildung 5: Verschiedene künstlich eingebrachte Fehlstellen in die synthetisch erzeugte Punktwolke. Diese konnten durch Parameter in ihrer Stärke kontrolliert werden.

Die beschriebenen Augumentierungsmethoden können zur Erhöhung der Datenvariationen und zur Verbesserung der Genauigkeit von Machine-Learning-Modellen beitragen.

Domain Randomization wurde angewendet, um die Trainingsdaten so zu variieren, dass der Realfall im besten Fall als eine Stichprobe aus der synthetischen Datenverteilung auftritt. Durch diese Variation sollte verhindert werden, dass das Modell sich nur auf spezifische Trainingsdaten beschränkt und nicht auf den Realfall übertragbar ist.

Die synthetischen Trainingsdaten wurden anschließend mit den realen Daten kombiniert, um ein umfangreiches Trainingsset zu erstellen. Dabei wurde darauf geachtet, dass die Balance zwischen den beiden Werkzeugteilen und ihren jeweiligen Merkmalen gewahrt bleibt.

Das Ergebnis dieses Arbeitspaketes ist eine Sammlung von Trainingsdaten für die Erkennung von Graten und Nocken auf den Werkzeugteilen mit Werkzeug-Nr. 36851 und 33000. Diese Trainingsdaten wurden durch die Verwendung von synthetischen Daten und Augumentierungstechniken vervollständigt und durch die Anwendung von Domain Randomization auf den Realfall übertragbar gemacht. Diese Daten werden in Arbeitspaket 4.2 verwendet, um verschiedene KI-Ansätze zu untersuchen.

AP4.2: Entwicklung der KI Algorithmen

In diesem Teilarbeitspaket waren die Projektpartner HS, BBS, AIT unter Koordination von HS aktiv. Es folgte eine enge Abstimmung mit AP4.1.

Vorsitzender des Vorstands:
Prof. Dr. Volker Nestle

Geschäftsführer:
Clemens Pecha

Institutsleiter:
Prof. Dr. Oliver Amft
Prof. Dr.-Ing. Alfons Dehé
Dr.-Ing. Karl-Peter Fritz
Prof. Dr. techn. Boris Mizaikoff
Apl. Prof. Dr. Felix von Stetten
Prof. Dr.-Ing. Roland Zengerle
Prof. Dr.-Ing. André Zimmermann

Sparkasse Schwarzwald-Baar
BIC: SOLADES1VSS
IBAN: DE82 6945 0065 0000 0927 26
USt-IdNr.: DE147807640

Seite 7 von 19

Definition der Entwicklungsumgebung

Bei der Definition der Entwicklungsumgebung hat sich das Team von BBS für einen GPU-Server mit der Linux-Distribution Ubuntu entschieden. Die Entscheidung zwischen Keras und PyTorch fiel zugunsten von Keras aus, da Keras bereits bei BBS im Einsatz ist.

Algorithmen Entwicklung zur Generierung von synthetischen Trainingsdaten

Bereits im frühen Stadium des Projekts wurde die starke Limitierung realen Trainingsdaten erkannt. Daher wurde früh der Fokus auf die Identifizierung von potentiellen KI-Einsatzmöglichkeiten gelegt, um dieses Problem der Limitierung zu beheben. Im Rahmen von AutoVikki wurden zwei Ansätze der Sim-to-Real-Domain-Adaptation zur Schließung der Lücke zwischen simulierter und realer Welt untersucht.

1. Der erste Ansatz basierte auf einem vortrainierten Sensornetz²³. Das Sensornetz lernt, wie künstliche Bilder so verändert werden können, dass sie den realen Bildern ähnlicher werden.
2. Der zweite Ansatz nutzt das CycleGAN-Netzwerk⁴⁵, um Bilder aus einer Domäne in eine andere zu transformieren.

Beide Ansätze wurden aufgrund mangelnder realer oder synthetischer Bilder aus Szenario B nicht nachtrainiert und das Projekt wurde aufgrund der Entscheidung zur Verwendung einer 3D-Kamera und der Verarbeitung von 3D-Bildinformationen eingestellt.

Da zum Zeitpunkt des Nachtrainierens des Netzes keine realen oder synthetischen Bilddaten aus Szenario B zur Verfügung standen, konnte keines der Netze im Rahmen des Projekts nachtrainiert werden. Nachdem die Entscheidung auf die Verwendung einer 3D-Kamera und die Verarbeitung von 3D-Bildinformationen gefallen ist, wurden die Arbeiten an Sim-2-Real für RGB bildbasierte Datenverarbeitung eingestellt. Dies ist dadurch begründet, dass die Simulation von 3D-Informationen sich auf die räumliche Geometrie und Textur des Zielobjekts beschränkt und die genaue Modellierung der Farbinformation, Farbbalance, Helligkeit und Sättigung vernachlässigt werden kann. Die Entscheidung, 3D-Punktwolken anstatt Sim-to-Real-Techniken zu verwenden, wurde nicht nur aufgrund der Anforderungen der vorliegenden Aufgabe getroffen. Hinzu kamen Schwierigkeiten und unzureichende Ergebnisse bei der Anwendung von Sim-to-Real-Techniken (siehe Abbildung 6). Die Techniken waren schwierig anzuwenden und es war schwierig, realistische Szenarien zu schaffen, die den realen Szenarien ähnelten. Insgesamt zeigt dieser Fall, dass die Wahl der richtigen Datenquelle von entscheidender Bedeutung ist, um erfolgreiche Ergebnisse zu erzielen.

² <https://arxiv.org/pdf/1809.06256.pdf>

³ <https://github.com/alexacarlson/SensorEffectTransferNetwork>

⁴ <https://arxiv.org/pdf/1703.10593.pdf>

⁵ <https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix>

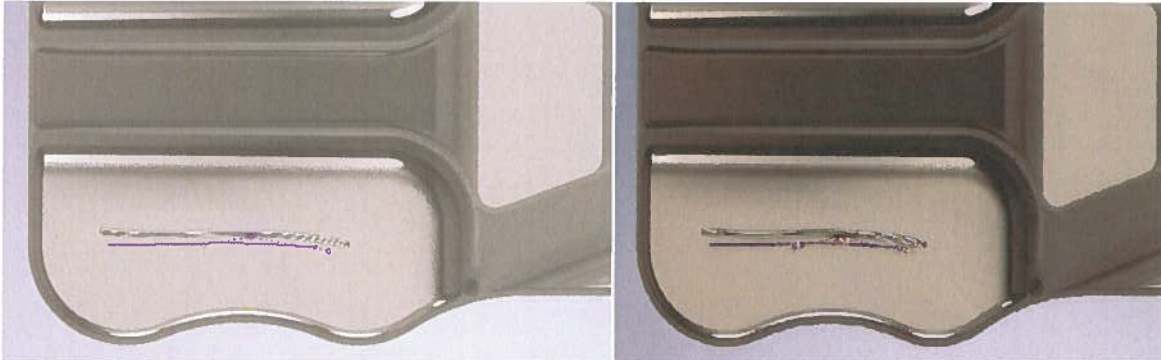


Abbildung 6: Auf der linken Seite ist ein Werkstück zu sehen, das auf Basis von CAD-Modellen erstellt wurde und absichtlich einen künstlich eingebrachten Fehler aufweist. Auf der rechten Seite ist die Ausgabe eines vorab trainierten GANs zu sehen, das auf Sim-2-Real-Daten trainiert wurde. Das GAN-Netzwerk war nicht in der Lage, hochwertige reale Bildmerkmale wie Texturen in die synthetischen Bilder einzufügen.

Stattdessen wurden synthetische Trainingsdaten auf Basis von CAD-Modellen generiert, welche durch Raytracing-Algorithmen erzeugt und von AIT an HS bereitgestellt wurden. Diese synthetischen Trainingsdaten wurden dann mithilfe von verschiedenen Techniken für 3D-Punktwolken-Daten augmentiert, um die Datenmenge zu erhöhen und die Robustheit des Modells zu verbessern.

Insgesamt lässt sich sagen, dass die Identifizierung von potentiellen KI-Einsatzmöglichkeiten eine komplexe Herausforderung darstellt, die einen umfassenden Ansatz erfordert. Die Verwendung von verschiedenen Techniken und Methoden kann dazu beitragen, die Wirksamkeit der Trainingsdaten zu verbessern und den Unterschied zwischen synthetischen und realen Daten zu minimieren. In diesem Projekt hat sich jedoch gezeigt, dass die Verwendung von Sim-2-Real Techniken für RGB-basierte Datenverarbeitung unzureichende Ergebnisse liefert. Stattdessen ist die Verwendung von 3D-Kameras und die Verarbeitung von 3D-Bildinformationen empfehlenswert, um die Wirksamkeit der Trainingsdaten zu verbessern.

Entwicklung der KI Algorithmen

Im Folgenden Abschnitt werden die Arbeiten zur Entwicklung der KI beschrieben.

Da im späteren Projektverlauf ausschließlich erhabene Fehler, nämlich Grat- und Nockenidentifikation, mit ausschließlich 3D-Eingabedaten ohne RGB-Bildinformation sowie die Erkennung der Fehlerausprägung betrachtet wurden, wurden die Ausgabeschichten der zu untersuchten Modelle so modifiziert, dass die allgemeine Problemstellung nicht mehr eine reine Semantische-Segmentation darstellt, sondern eine 2D-Regression. Das bedeutet, dass die Ausgabe eine 2D-Bildmatrix ein einziger Kanal ist, der sowohl die Lokalisierung anhand der Pixelposition als auch die Ausprägung des erhabenen Fehlers anhand der entsprechenden Werte hinter den Pixeln enthält.

Die untersuchten Netzarchitekturen für das Szenario B:

1. Mask R-CNN⁶, da dieses bereits bei BBS in Szenario A zum Einsatz kommt,
2. DeepLab-V3⁷ mit ResNet 50 und ResNet 101 Backbone,
3. U-Net⁸ und

⁶ https://github.com/matterport/Mask_RCNN

⁷ <https://github.com/leimao/DeepLab-V3>

⁸ <https://github.com/karolzak/keras-unet>

4. PointNet++⁹

Im Rahmen des Szenario B wurde eine umfassende Evaluierung von verschiedenen Segmentierungs-Netzarchitekturen durchgeführt, um die Eignung für die Erkennung erheblicher Fehler bei Werkzeugteilen zu prüfen. Dabei wurde auch das Mask R-CNN Modell hinsichtlich seiner Tauglichkeit geprüft und aufgrund der anwendungsbezogenen Anforderungen als zu komplex und aufwendig zu trainieren bewertet. Insbesondere führten eine zu hohe Datenanforderung des Trainings, ein zu hoher Rechenaufwand aufgrund von hoher Architekturkomplexität sowie Schwierigkeiten bei der Segmentierung von kleineren Objekten zu einem Aus für das Mask R-CNN Modell. In Anschluss wurden weitere Segmentierungs-Netzarchitekturen untersucht, darunter das DeepLabv3-Modell mit zwei unterschiedlichen Backbones (ResNet 50 und ResNet 101) sowie die einfach gehaltene U-Net-Architektur. Ein weiteres untersuchtes Modell war das PointNet++-Modell, das speziell für die Verarbeitung von 3D-Punktwolken konzipiert wurde. Auch bei diesem Modell wurde die Ausgabe so modifiziert, dass die Ausgabe des Modells einer 2D-Regressionskarte entspricht. Den ganzen Modellen wurde ein alternatives Vorgehen gegenübergestellt. Ein 2D-Regressionsmodell von Scikit-Learn, welches die Bodenplatte der Werkzeugteile modelliert. Auf Basis dieses trainierten Regressionsmodells konnte die Orientierung des Objekts im 3D-Raum bestimmt werden und Transformationen an der Punktwolke durchgeführt werden, sodass der Grad bzw. der Nocken samt seiner Ausprägung direkt aus der Punktwolke abgelesen werden konnte.

Das genannte Vorgehen und Modellen wurden auf den synthetischen 3D Trainingsdaten trainiert und an realen Testdaten evaluiert und die Ergebnisse miteinander verglichen. Schnell zeigte sich, dass bei der Verwendung von neuronalen Netzen für die Extraktion von Tiefenwerten aus 3D-Punktwolken Daten Vorsicht geboten ist, da die Eingabe mit Gewichten multipliziert wird und dadurch verfälscht wird. Dadurch wurden die gewünschten Tiefenbildinformationen verändert und bei Vorhersagen auf ungesicherte Grat- und/oder Nockenausprägungen war die Ausgabe der KI-Vorhersage verzerrt und damit unzureichend. Im vorliegenden Fall von Szenario B wurde der Einsatz von neuronalen Netzen zur Bewältigung des Problems als nicht hilfreich bewertet. Damit wurde das performanteste und genaueste Modell mit dem 2D-Regressionsmodell von Scikit-Learn bestimmt.

Insgesamt hat sich gezeigt, dass die Anpassung der Ausgangsschichten der Modelle an die spezifischen Anforderungen des Projekts einen entscheidenden Einfluss auf die Leistungsfähigkeit der Modelle hat. Die 2D-Regression bietet dabei eine effektive Alternative zur modifizierten Semantischen-Segmentierung für die Erkennung erheblicher Fehler bei den untersuchten Werkzeugteilen. Das 2D-Regressionsmodell von Scikit-Learn zur Modellierung der Bodenplatte erwies sich als das performanteste und genaueste Modell für die Erkennung von erheblichen Fehlern auf Werkzeugteilen. Die Implementierung des Modells erforderte jedoch eine umfassende Vor- und Nachverarbeitung der 3D-Eingabedaten, um die Bodenplatte der Werkzeugteile zu modellieren und die Orientierung der Werkstücke im 3D-Raum zu bestimmen.

Flexibilität: Flower ermöglicht die Integration von verschiedenen maschinellen Lern-Frameworks wie TensorFlow, PyTorch oder Keras. Die Cloud-Umgebung sollte daher die Integration dieser Frameworks unterstützen und auch die Möglichkeit bieten, eigene Frameworks zu verwenden.

AP4.3: Training, Validierung und Test der KI Algorithmen

In diesem Teilarbeitspaket waren die Projektpartner HS und AIT unter Koordination von HS aktiv. Es folgte eine enge Abstimmung mit AP4.2.

⁹ <https://github.com/dgriffiths3/pointnet2-tensorflow2>

Um die Vorverarbeitung der Daten für eine 3D-Punktwolke zu automatisieren, wurde ein vollständiges Python-Framework entwickelt, das eine Verbindung zur 3D-Kamera herstellt und die Punktwolke zunächst vorverarbeitet. Die Vorverarbeitung besteht aus drei Schritten: Zuschneiden der Punktwolke, um das Werkzeugteil hervorzuheben, Entfernen des Hintergrunds um das Werkzeugteil freizustellen und eine Zentrierung der Punktwolke. Anschließend wurde mittels Scikit-Learn die Bodenplatte mittels eines 2D Regressionsmodells modelliert, um die Orientierung des Bauteils im Raum zu bestimmen. Postverarbeitungsschritte umfassten das Rotieren der 3D-Punktwolke im Raum, sodass die Bodenplatte konstant in der XY-Dimension lag, das Invertieren und verschieben des Bauteils in der Z-Dimension, um das Bauteil im 3D-Raum so zu platzieren, dass die Bodenplatte eben in der XY-Dimension mit $Z=0$ lag. Dadurch war die Ausprägung des erhabenen Fehlers in positive Z-Richtung gewährleistet. Als letzter Bearbeitungsschritt, wurden die 2D Regressionskarte der Nocken bzw. der Grate des Werkzeugteils herausgestellt und in einem 3x3-Array bzw. 5x6-Array segmentiert. Dieses Array wurde schließlich als Grundlage für die Erstellung von Schleifanweisungen verwendet.

Dieses Vorgehen hat sich als äußerst effektiv erwiesen. Das Framework nutzt die Bibliotheken NumPy, Pandas, Zivid Python und Scikit-Learn. Die Vorverarbeitung der Daten war ein wichtiger Schritt, um die Genauigkeit und Performance des beschriebenen Vorgehens zu verbessern. Das vollständige Framework wurde AIT bereitgestellt sowie eine Unterweisung haben stattgefunden. Das trainierte 2D Regressionsmodell wurde erfolgreich zur Erkennung von erhabenen Fehlern auf Werkzeugteilen eingesetzt. Das Modell erwies sich als robust gegenüber verschiedenen Werkstückgrößen und -formen und konnte zuverlässig erhabene Fehler auf Werkzeugteilen erkennen

Ergebnisse

Die Abbildung 7 (Werkzeugteils 33000) und 8 (Werkzeugteils 36851) beschreibt den Prozess der Verarbeitung einer Aufnahme. Das Bild wurde als 3D Punktwolke aufgenommen und als Ausgangsbasis für verschiedene Bearbeitungsschritte verwendet. Die detaillierten Schritte zur Erkennung der erhabenen Fehler und Übermittlung an die Schleifanweisungen werden in Abbildung 7 (Nocken) und Abbildung 8 (Grat) veranschaulicht. Diese Abbildungen zeigt die Z-Information als Tiefenbild aus der originalen Aufnahme des Werkzeugteils 33000 und 36851. Der Hintergrund wurde entfernt, um eine bessere Vorbereitung für die 2D-Regression zum Modellieren der Bodenplatte zu ermöglichen. Nach der Rotation und Transformation der Punktwolke auf Basis des 2D Regressionsmodells wurde der Nocken durch weitere Transformationen herausgestellt. Schließlich erfolgte die Segmentierung in ein 3x3 Array bzw. 5x6 Array, das als Information zur Schleifanweisung diente. Abbildung 9 zeigt die Schleifanweisung für einige weitere reale Bildaufnahmen. Abbildung 10 zeigt die Schleifanweisung des Grates für ein Werkzeugteil, dass sich von Werkzeugteil 36851 geometrisch unterscheidet.

Diese Schritte zeigen, wie die Verarbeitung von Daten durch Machine-Learning-Modelle eine genaue und automatisierte Erkennung von Fehlern ermöglicht, um eine präzise Schleifanweisung zu generieren.

Vorsitzender des Vorstands:
Prof. Dr. Volker Nestle

Geschäftsführer:
Clemens Pecha

Institutsleiter:
Prof. Dr. Oliver Amft
Prof. Dr.-Ing. Alfons Dehé
Dr.-Ing. Karl-Peter Fritz
Prof. Dr. techn. Boris Mizaikoff
Apl. Prof. Dr. Felix von Stetten
Prof. Dr.-Ing. Roland Zengerle
Prof. Dr.-Ing. André Zimmermann

Sparkasse Schwarzwald-Baar
BIC: SOLADES1VSS
IBAN: DE82 6945 0065 0000 0927 26
UST-IdNr.: DE147807640

Seite 11 von 19

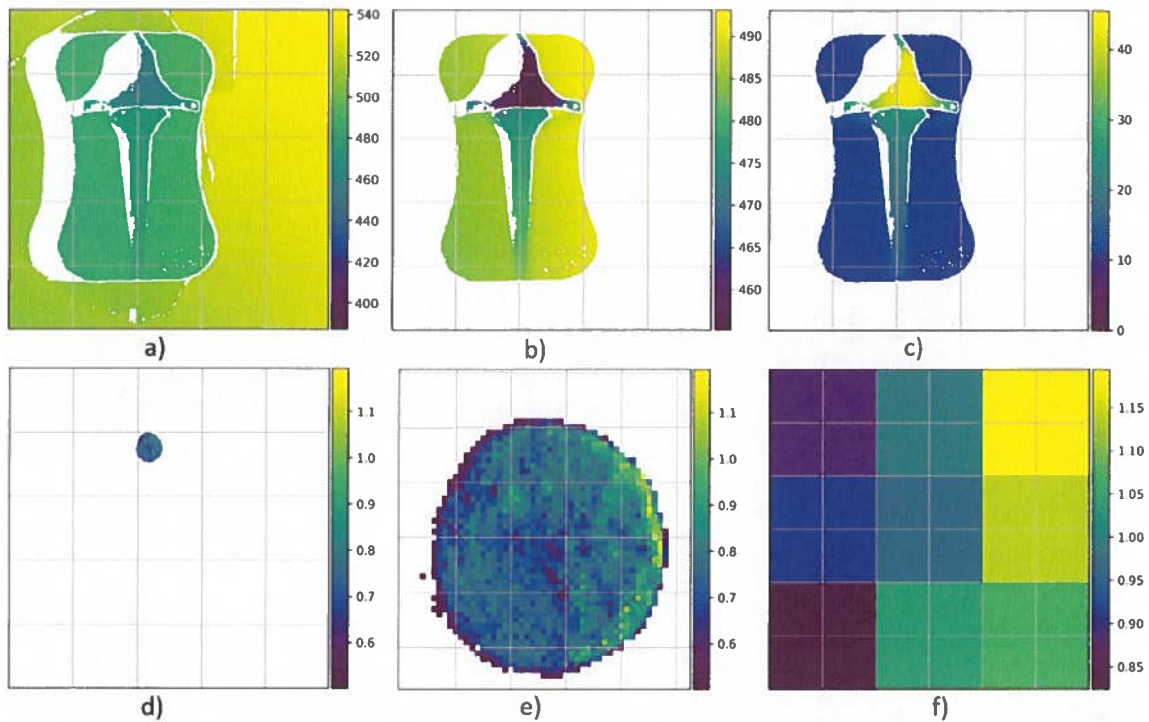


Abbildung 7: a) Die z-Information als Tiefenbild aus der originalen Aufnahme gezeigt für das Werkzeugteil 33000. b) Mit entfernten Hintergrund als Vorbereitung für die 2D-Regression zum Modellieren der Bodenplatte. c) Nach Rotation und Transformation der Punktwolke auf Basis des 2D Regressionsmodells. d) Herausstellen des Nockens durch weitere Transformationen. e) Segmentierung in ein 3x3 Array als Information zur Schleifanweisung.

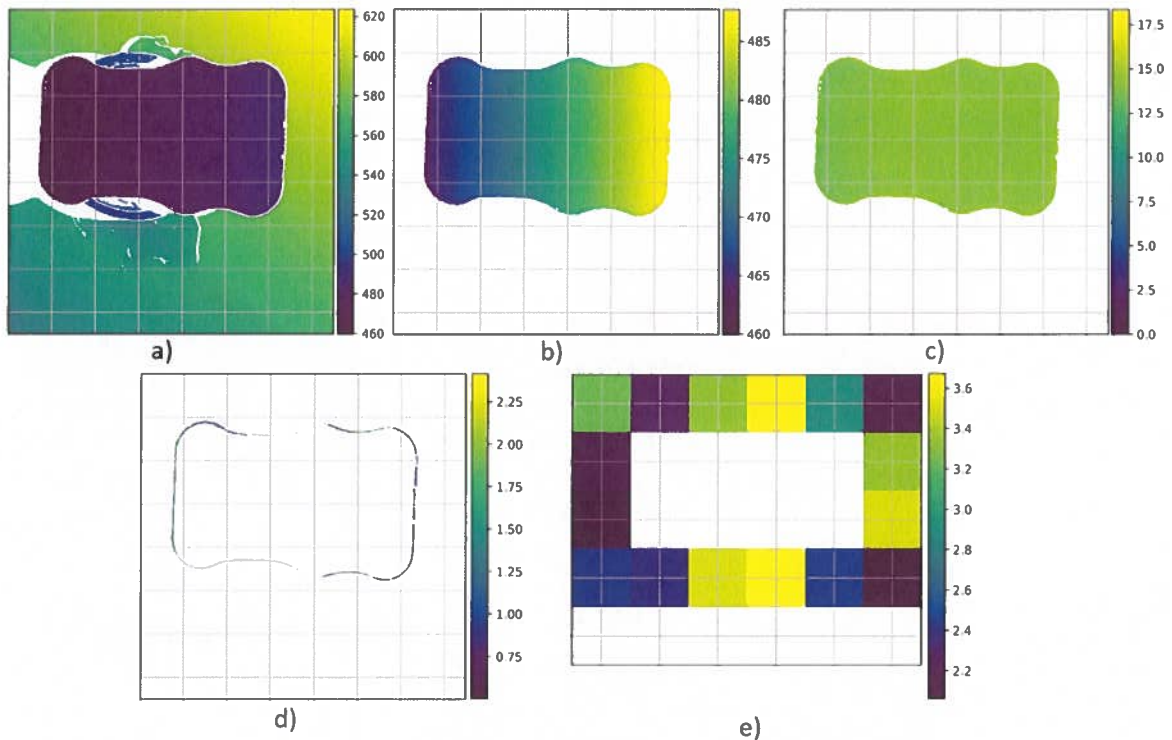


Abbildung 8: a) Die z-Information als Tiefenbild aus der originalen Aufnahme gezeigt für das Werkzeugteil 36851. b) Mit entferntem Hintergrund als Vorbereitung für die 2D-Regression zum Modellieren der Bodenplatte. c) Nach Rotation und Transformation der Punktwolke auf Basis des 2D Regressionsmodells. d) Herausstellen des Grates durch weitere Transformationen. e) Segmentierung in ein 5x6 Array als Information zur Schleifanweisung.

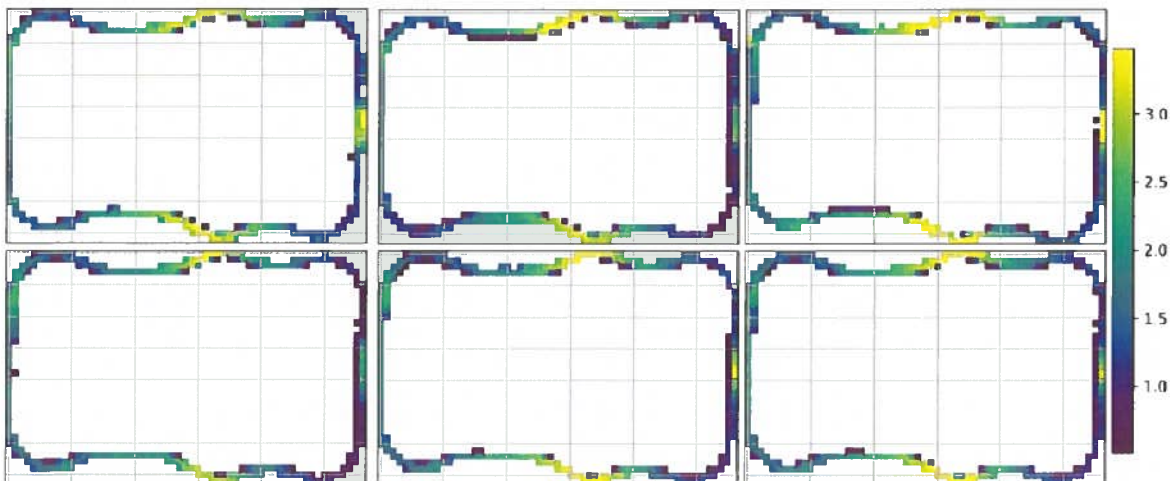


Abbildung 9: Sechs weitere Beispiele von realen Aufnahmen des Werkzeugteils 36851. Der gezeigte Grat variiert zwischen 0 und 3,5mm gezeigt mit einer 40x100 Segmentierung.

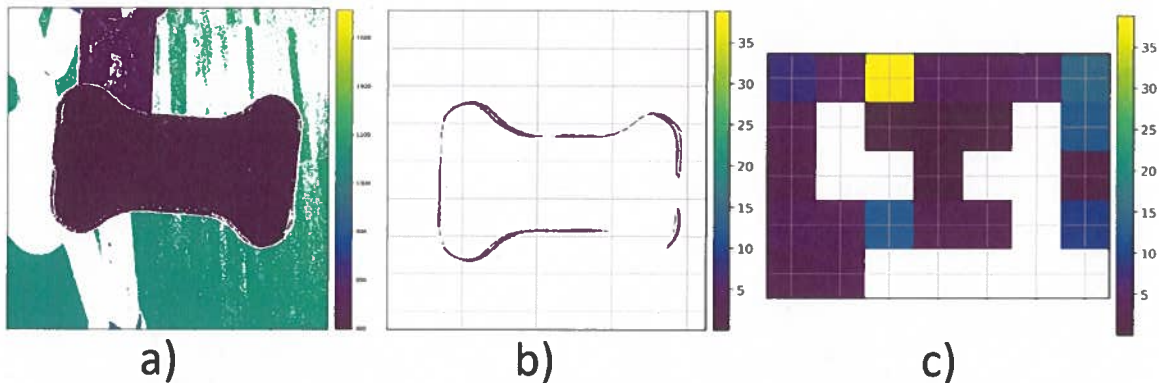


Abbildung 10: Die z-Information als Tiefenbild aus der originalen Aufnahme gezeigt für ein sich von Werkzeugteil 36851 geometrisch unterscheidbares Objekt. b) Ausgabe der Vorverarbeiteten und transformierten Punktwolke auf Basis der 2D-Regression. c) Segmentierung in ein 5x6 Array als Information zur Schleifanweisung.

Verteiltes Lernen über verschiedene Roboterzellen durch die Nutzung von Federated Learning

Ein Federated Learning Framework soll für das Roboterzellenübergreifende Training umgesetzt werden (verteiltetes Lernen). Zur Bewertung der Kompatibilität mit den Projektanforderungen wurden die spezifischen Anforderungen des Projekts identifiziert und für jedes Framework analysiert. Es wurden verschiedene Faktoren berücksichtigt, wie beispielsweise die Unterstützung von bestimmten Modelltypen, die Skalierbarkeit des Frameworks und die Möglichkeit zur Integration in vorhandene Systeme. Um die Leistung der Frameworks zu bewerten, wurden verschiedene Benchmarks durchgeführt, um die Geschwindigkeit des Trainings und Inferenz sowie den Ressourcenbedarf zu bestimmen. Die Flexibilität der Frameworks wurde anhand der Fähigkeit zur Verarbeitung verschiedener Datenformate und -strukturen sowie der Unterstützung verschiedener Modelltypen und -architekturen bewertet. Es wurde auch berücksichtigt, wie einfach es war, das Framework an spezifische Anforderungen anzupassen und zu erweitern. Um den Stand der Technik der Frameworks zu bewerten, wurden verschiedene Publikationen analysiert, um die neuesten Entwicklungen und Fortschritte in der Forschungsgemeinschaft zu untersuchen. Darüber hinaus wurde auch die Fähigkeit der Frameworks zur Integration mit anderen Technologien und Tools bewertet. Die Bedienbarkeit der Frameworks wurde anhand der Benutzerfreundlichkeit und der Einfachheit bei der Installation und Verwendung des Frameworks bewertet. Ein weiterer Faktor der berücksichtigt wurde war, wie einfach es war, die Dokumentation und Unterstützung für das Framework zu finden und zu nutzen. Nach eingehender Analyse und Bewertung der verschiedenen Federated Learning Frameworks wurde Flower¹⁰ als das beste Framework bewertet. Dies liegt hauptsächlich an seiner einfachen Bedienbarkeit und der Tatsache, dass es auf modernste Technologien wie Protobuf und gRPC aufbaut. Darüber hinaus ist Flower auch sehr flexibel und kann einfach an spezifische Anforderungen angepasst werden.

Konkret handelte es sich dabei um die Verwendung der KI-Architektur Mask R-CNN für die Bilderkennung in einem verteilten, privaten Umfeld. Die Anpassung von Flower erfolgte durch die Erstellung einer angepassten Version von Mask R-CNN zum Ein- und Auslesen der Gewichte und dem Training des Modells auf einer lokal verfügbaren Datensatzkopie. Nach erfolgreicher Anpassung von Flower wurden sowohl das Framework als auch die angepasste Version von Mask R-CNN an

¹⁰ <https://arxiv.org/pdf/2007.14390.pdf>

BBS bereitgestellt. Hierbei wurde eine Unterweisung durchgeführt, um sicherzustellen, dass das Framework und die angepasste Architektur effektiv nutzen und verwalten können.

Vergleich zur ursprünglichen Vorhabensbeschreibung

Der Vergleich mit der ursprünglichen Vorhabensbeschreibung zeigt, dass die meisten geplanten Aufgaben umgesetzt wurden. In Szenario B wurden 400 künstliche Punktwolke-Daten auf Basis eines CAD-Modells für beide Fehlerarten generiert. Diese wurden genutzt, um die verschiedenen Deep-Learning-Ansätze in AP4.2 zu vergleichen. Zusätzlich wurden in Szenario B für beide Fehlerarten jeweils 30 reale Punktwolke-Daten erzeugt, die in AP4.3 genutzt wurden, um die Fehlererkennung der Deep-Learning-Methoden aus AP4.2 zu evaluieren. Da sich für einen vollausgestatteten GPU-Server entschieden wurde, der in die Roboterzelle integriert wurde, wurde das Framework nicht wie ursprünglich vorgesehen auf den Einsatz eines Embedded-Geräts optimiert. Stattdessen wurde ein Federated-Learning-Framework entwickelt und die dazu notwendige Cloud-Umgebung geschaffen. Das verteilte Lernen mittels Federated Learning Framework wurde mit der finalen KI nicht implementiert, da es für das 2D-Regressionsmodell keine Vorteile aus Sicht der Anwendung ergibt.

In AP4.3 wurden verschiedene Deep-Learning-Ansätze untersucht. Dabei hat sich herausgestellt, dass KI-Modelle auf Basis von Segmentierungsansätzen Schwierigkeiten hatten, die Fehler inklusive ihrer Ausprägung zu modellieren. Es wurde ein 2D-Regressionsmodell entwickelt, das nicht versucht, den Fehler inklusive Ausprägung zu modellieren, sondern die Bodenplatte der Bauteile und auf Vorhersage dieses Modells Transformationen anwendet, um die Punktwolke so zu transformieren, dass der Fehler direkt aus den Daten gelesen werden kann. Demnach ist das in dem Projekt entwickelte Debugging-Tool für Erklärbare KI nur im Rahmen der Entwicklung umgesetzt und getestet worden. Es kam aber nicht zum finalen Einsatz, wie in der ursprünglichen Vorhabensbeschreibung vorgesehen.

Insgesamt hat diese Forschungsarbeit gezeigt, dass die Erkennung von erheblichen Fehlern auf Werkzeugteilen mit Hilfe von 3D-Punktwolken und Regressionsmodellen eine vielversprechende Methode ist. Die Verwendung von 2D-Regressionsmodellen anstelle von Semantische-Segmentation-Modellen bietet den Vorteil einer höheren Genauigkeit und einer schnelleren Verarbeitung von Daten. Darüber hinaus bietet die Verwendung von 3D-Punktwolken den Vorteil einer robusten und zuverlässigen Erfassung von Werkstückgeometrien und -formen. Die Ergebnisse dieser Forschungsarbeit können in der Fertigungsindustrie zur Qualitätskontrolle von Werkzeugteilen eingesetzt werden. Die vorgestellte Methode bietet ein schnelles und zuverlässiges Verfahren zur Erkennung von erheblichen Fehlern auf Werkzeugteilen, was dazu beitragen kann, die Produktionszeit zu verkürzen und die Qualität von Werkstücken zu verbessern

AP5 Konzeptionierung, Implementierung und Validierung des Vision Systems

AP5.1: Kameraauswahl, Kalibrierung

Im Rahmen dieses Teilarbeitspaketes wurde HS trotz ursprünglicher Nichtteilnahme mit der Aufgabe betraut, die Zivid One+ Kamera zu kalibrieren. Das Ziel bestand darin, die Reflexionen auf stark reflektierenden Werkzeugoberflächen zu reduzieren. Dabei wurden verschiedene Einstellungen wie Belichtungszeit, Blendenöffnung, Helligkeit und Verstärkung untersucht. Durch systematische Untersuchungen konnten geeignete Einstellungen identifiziert werden, die die Bildqualität verbesserten und die Genauigkeit der 3D-Bilder erhöhten.

AP5.2: Aufbau eines Messtands zum Sammeln von Trainingsdaten für Szenario B

Im Rahmen dieses Teilarbeitspaketes wurde HS trotz ursprünglicher Nichtteilnahme mit der Aufgabe betraut, OFK beim Aufbau eines Messtandes zu unterstützen. Die Unterstützung bestand darin, eine

geeignete Kameraposition relativ zu den Werkzeugteilen zu finden, welche die Reflektion minimiert. Dafür wurden verschiedene Aspekte berücksichtigt. Die Positionierung der Kamera wurde so gewählt, dass sie einen optimalen Blickwinkel auf die Werkstücke ermöglichte. Dabei wurden Faktoren wie Abstand, Neigung und Blickrichtung sorgfältig abgestimmt, um eine gute Erfassung der Werkzeugoberflächen zu gewährleisten. Auch wurden verschiedene Lichteinflüsse sowie deren Abschirmungen im Rahmen des Messstandaufbaus untersucht. Ziel war es, potenzielle Störungen durch externe Lichtquellen zu minimieren und eine zuverlässige Erfassung der Werkzeugoberflächen sicherzustellen.

Der optimierte Messstandaufbau ermöglichte somit eine zuverlässige und präzise Erfassung der Werkzeugoberflächen in Szenario B und legte die Grundlage für weitere Analysen und Auswertungen im Rahmen dieser Forschungsarbeit.

AP5.3: Entwicklung der Vision Algorithmen

Dieses Teilarbeitspaket konnte nicht wie vorgesehen von BBS vollständig bearbeitet werden. In Szenario B zeigte sich, dass die von BBS vorbereitete 3D-Kamera für die Aufnahme von Bildern bei Otto Fuchs ohne weitere Änderungen nicht gut nutzbar war. Da jedoch der zuständige und erfahrene Mitarbeiter mittlerweile BBS verlassen hatte, erhielt HS die 3D-Kamera und GPU Server von Otto Fuchs zugesendet, um die Vision Algorithmen zu überarbeiten.

Zunächst wurde die 3D-Kamera erfolgreich in Betrieb genommen und es wurden die erforderlichen Treiber und Software-Pakete auf dem GPU-Server installiert. Im nächsten Schritt wurden die Kameraeinstellungen auf die beiden Werkzeugteile angepasst und optimale Einstellungen gesucht, um die Reflexionen zu minimieren. Um die spätere Anwendung in der Roboterzelle zu erleichtern, wurden die von OFK erhaltenen Testwerkzeugteile genutzt, um Positionierungsempfehlungen zu ermitteln und festzuhalten. Dabei wurde insbesondere darauf geachtet, dass die Positionierung präzise und zuverlässig ist, um Reflektionen zu minimieren und eine effektive automatisierte Produktion zu ermöglichen. Diese wurden mit einem Abstand von ca. 350-400mm der Bodenplatte zur Kamera mit einer Neigung von 5-10° relativ zur Kamera bestimmt. Um Trainingsdaten bei OFK aufzunehmen, wurde mit Python ein Programm geschrieben, das eine vollständige Aufnahme bestehend aus RGB- und 3D-Punktwolke aufnimmt und speichert. Dieses Framework wurde getestet und an OFK bereitgestellt, um sicherzustellen, dass es den Anforderungen entspricht. Später wurde dieses Framework in das vollständige Framework aus AP4.3 integriert, um eine nahtlose und effiziente Verwendung zu gewährleisten.

AP5.4: 3D Modelle

HS war nicht Teil dieses Teilarbeitspaketes.

AP6 Konzeptionierung, Implementierung und Validierung der Roboterzelle

AP6.1: Entwicklung einer zentralen Kommunikationsplattform

HS war nicht Teil dieses Teilarbeitspaketes.

AP6.2: Entwicklung der herstellerunabhängigen Schnittstellenebene

HS war nicht Teil dieses Teilarbeitspaketes.

AP6.3: Entwicklung der Datenbank Cloud Umgebung aus AP3.4

HS war nicht Teil dieses Teilarbeitspaketes.

Vorsitzender des Vorstands:
Prof. Dr. Volker Nestle

Geschäftsführer:
Clemens Pecha

Institutsleiter:
Prof. Dr. Oliver Amft
Prof. Dr.-Ing. Alfons Dehé
Dr.-Ing. Karl-Peter Fritz
Prof. Dr. techn. Boris Mizaikoff
Apl. Prof. Dr. Felix von Stetten
Prof. Dr.-Ing. Roland Zengerle
Prof. Dr.-Ing. André Zimmermann

Sparkasse Schwarzwald-Baar
BIC: SOLADES1VSS
IBAN: DE82 6945 0065 0000 0927 26
UST-IdNr.: DE147807640

AP6.4: Entwicklung der Werkzeuge und des digitalen Zwillings für Szenario B

HS war nicht Teil dieses Teilarbeitspaketes.

AP6.5: Aufbau der Roboterzelle für Szenario B

HS war nicht Teil dieses Teilarbeitspaketes.

AP7 Integration der KI und Vision Systeme in die Roboterzelle

AP7.1: Szenario A: Automatisierte In-vitro Pflanzenvermehrung

HS war nicht Teil dieses Teilarbeitspaketes. Im Verlauf des Projekts wurde festgestellt, dass Szenario A von BBS ohne Beteiligung der anderen Projektpartner durchgeführt wurde. Diese Entscheidung von BBS hat sich als problematisch erwiesen, da die anderen Projektpartner nicht in der Lage waren, ihre Expertise und ihr Know-how einzubringen, um sicherzustellen, dass die Durchführung von Szenario A in Übereinstimmung mit den Projektzielen und -anforderungen erfolgte. Die entsprechenden Ergebnisse finden sich im Bericht der BBS.

AP7.2: Szenario B: Automatische Nachbearbeitung geschmiedeter Bauteile

In diesem Teilarbeitspaket waren die Projektpartner HS, AIT und OFK unter Koordination von AIT aktiv. Es folgte eine enge Abstimmung mit AP4 – AP6. Im Rahmen dieser Forschungsarbeit wurde eine Roboterzelle für die automatisierte Bearbeitung von Werkstücken konzipiert und realisiert. Ein zentraler Bestandteil der Steuerungssoftware für die Roboterzelle ist das Python-Framework, das von HS entwickelt wurde. Dieses Framework basiert auf den Modulen AP4.3 und AP5.3 und bildet die Grundlage für die automatisierte Steuerung der Roboterzelle. HS stellte das vollständige Python-Framework für die Robotersteuerung zur Verfügung. Dabei wurde besonderer Wert auf eine einfache und intuitive Handhabung gelegt, um den Einsatz der Roboterzelle für den Anwender möglichst einfach zu gestalten. Das Framework ist modular aufgebaut und ermöglicht die Anpassung an die individuellen Anforderungen der Roboterzelle. Bei der Inbetriebnahme der Roboterzelle AIT von HS unterstützt. Hierbei wurden insbesondere AIT in der Handhabung des Python Frameworks geschult. Durch die modulare Struktur des Frameworks war es möglich, individuelle Anpassungen an der Steuerungssoftware vorzunehmen und so eine optimale Integration in den Produktionsprozess zu gewährleisten.

AP8 Systemtests der Roboterzellen

AP8.1: Szenario A

HS war nicht Teil dieses Teilarbeitspaketes. Siehe AP7.1

AP8.2: Szenario B

Im Rahmen dieser Forschungsarbeit war geplant, einen Langzeittest mit den entwickelten Komponenten durchzuführen. Aufgrund von verschiedenen Faktoren konnte dieser Test jedoch nicht wie geplant durchgeführt werden. Zunächst führte die Corona-Pandemie zu Verzögerungen bei der Beschaffung der 3D-Kamera. Da viele Unternehmen und Lieferanten von der Pandemie betroffen waren, kam es zu Lieferengpässen und Verzögerungen bei der Produktion und Lieferung von Elektronikkomponenten, einschließlich der 3D-Kamera. Dies führte zu einer Verzögerung im Projektzeitplan, die nicht aufzuholen war. Schließlich musste die Forschungsarbeit auch mit dem Ausscheiden von externen Projektmitarbeitern und -partner zurecht kommen. Dies führte zu Verzögerungen und Engpässen in der Durchführung verschiedener Teilarbeitspakete. Obwohl HS in anderen Teilarbeitspaketen engagiert war, gab es keine Beteiligung an der Planung und Durchführung des Langzeittests. Dies war in erster Linie auf die oben genannten Faktoren zurückzuführen, die die Durchführung des Tests erschwerten oder unmöglich machten.

2. Der voraussichtliche Nutzen, insbesondere die Verwertbarkeit des Ergebnisses – auch konkrete Planungen für die nähere Zukunft im Sinne des fortgeschriebenen Verwertungsplans

Im Vorhaben AutoVikki wurden autonome visuelle Roboterinteraktionen durch KI an zwei Szenarien, nämlich Szenario A für die autonome in vitro Pflanzenvermehrung und Szenario B für die autonome Bearbeitung von geschmiedeten Gußteilen (*Door Stops*) erfolgreich erforscht. HS hat erfolgreich im Rahmen seines Teilprojektes den Schwerpunkt auf die Erforschung und Entwicklung von *künstlicher Intelligenz* unter Verwendung eines *Vision-Systems* auf Basis von Punktwolkendaten bei heterogenen Objekten gelegt. Mit den beiden beispielhaft entwickelten Szenarien ergibt sich für alle Konsortialpartner in dem jeweiligen Fachgebiet ein breites Anwendungsfeld für die entwickelten Technologien für sämtliche Industrieproduktionen, für die bisher aufgrund der Individualität und dadurch der Diffizilität einzelner Werkstücke Handarbeit notwendig war.

Durch den erfolgreichen Einsatz von KI und eines Vision-Systems zur Bearbeitung heterogener Objekte konnte in diesem Projekt eindrucksvoll demonstriert werden, dass die Produktion, die aufgrund von Kostengründen ins Ausland verlagert wurde, mithilfe dieser entwickelten Technologie wieder ins Inland geholt werden kann. Aufgrund dieser Tatsache sind die wirtschaftlichen Erfolgsaussichten für OFK und BBS in den Szenarien äußerst vielversprechend.

3. Während der Durchführung des Vorhabens dem Zuwendungsempfänger bekannt gewordenen Fortschritt auf dem Gebiet des Vorhabens auf anderen Stellen

HS sind während der Durchführung des Vorhabens keine Fortschritte auf dem Gebiet des Vorhabens bei anderen Stellen bekannt geworden.

4. Erfolgte oder geplante Veröffentlichungen des Ergebnisses

Der Abschlussbericht des Projekts ist auf der offiziellen Webseite von Hahn-Schickard unter der Rubrik "Publikationen" verfügbar. Zusätzliche Veröffentlichungen zu den Forschungsergebnissen sind in Planung.

Villingen-Schwenningen, 27.06.23

Ort, Datum

Hahn-Schickard-Gesellschaft
für angewandte Forschung e.V.
Wilhelm-Schickard-Str. 10
78052 Villingen-Schwenningen


i.V. 
Rechtsverbindliche Unterschrift und
Stempel des Zuwendungsempfängers

Vorsitzender des Vorstands:
Prof. Dr. Volker Nestle

Geschäftsführer:
Clemens Pecha

Institutsleiter:
Prof. Dr. Oliver Amft
Prof. Dr.-Ing. Alfons Dehé
Dr.-Ing. Karl-Peter Fritz
Prof. Dr. techn. Boris Mizaikoff
Apl. Prof. Dr. Felix von Stetten
Prof. Dr.-Ing. Roland Zengerle
Prof. Dr.-Ing. André Zimmermann

Sparkasse Schwarzwald-Baar
BIC: SOLADES1VSS
IBAN: DE82 6945 0065 0000 0927 26
UST-IdNr.: DE147807640