

# Haozheng Lin

---

Anschrift: Lillweg 13, 80939 München  
Telefon: +49 1625646803  
E-Mail: [linhaozheng@yahoo.com](mailto:linhaozheng@yahoo.com)  
Sprachen: Deutsch und Englisch  
Portfolio: <https://heizie.github.io>  
Github: [https://github.com/heizie/example\\_code](https://github.com/heizie/example_code)

## Berufliche Erfahrung

---

09.2018 – 10.2020      Werkstudent bei Bleenco GmbH  
Meine Aufgaben waren:

- Teilnahme am Projekt zur Überwachung der PSA-Bekleidung; Beitrag zu den Entwicklungs-Schwerpunkten
- Klassische Algorithmen im Bereich Computer-Vision (Deep-Learning-basiert) implementieren, modifiziert, und in einem Docker-Container gekapselt

Eigenständiges Projekt:

- Entwicklung eines Lablingswerkzeugs zur Erleichterung der Arbeit mit mobilen Geräten

## Ausbildung

---

09.2018 – 10.2021      Master Maschinenwesen  
(Regelungstechnik und Computer Vision)  
An der Technische Universität München

09.2014 – 09.2018      Bachelor Maschinenbau (Elektromobilität)  
An der Technische Universität Bergakademie Freiberg

09.2010 – 08.2014      Bachelor Germanistik  
An der Zengcheng Südchinesische Normale Universität

## Projekt-Erfahrung

---

04.2021 – 10.2021      **Masterarbeit:**  
Instanzsegmentierung zur Anwendung auf deformierbare lineare Objekte (**Note: 1,3**)

- Entwicklung von Anker-Relevanten-Modulen für den 4-dimensionalen RoI-Transformer von Grund auf
- Aufbau von angenommenen Modellen entsprechend den Fehlerfällen und Einsatz der

Optimierung in Bezug auf den Anwendungsszenario

- Entwurf und Verwaltung von Datensätzen; Schnelle Implementierung und Fehleranalyse von verschiedenen State-of-the-Art-Modellen als Grundlage
- Pytorch basierte Detectron 2 und MMDetection werden verwendet

04.2020 – 11.2020

#### **Semesterarbeit:**

Bildbasiertes Tracking von Instrumenten eines laparoskopischen Manipulatorsystems

- Aufbau einer echtzeitigen U-Net-basierten Modell zur Erkennung der Pose von medizinischen Instrumenten
- Deren Unterstützungsalgorithmen beschleunigen
- Modells auf FP16 mit TensorRT transformieren und durch ein C++-Skript geladen
- Geschwindigkeit: von 8,7 fps auf 44 fps optimiert
- Genauigkeit: 25 % weniger Pixelfehler und etwa 10 % höhere Recall- und Präzisionsrate
- Versuchsmäßige Sensorfusion für eine 3D-Rekonstruktion
- Tensorflow und TensorRT (C++) werden verwendet

08.2020

Praktikum: Autonomous Applications (Note: 1,7)

- ROS-basiertes Projekt
- Trajektorie verfolgen mit PID und simulierten Lidar-Daten
- Unfallvermeidung durch Geschwindigkeit in normale Richtung zur Verhinderung

## Kenntnisse

---

Software:

Python, C++, Matlab  
Tensorflow, Pytorch, MMDetection, Detectron2, TensorRt  
Docker, Solidworks

Sprachen:

Deutsch fließend in Wort und Schrift  
Englisch fließend in Wort und Schrift  
Chinesische-Muttersprache

Sonstige:

Führerschein Klasse B

## Interessen

---

Hobbys

Mikrokontroller programmieren (ESP8266, Arduino),  
3D Printer, FPV (Drohnen)