Osama Altalmas

Robotics & Embedded Systems Engineer · ROS 2 · Navigation · Manipulation · Bahnplanung · Perzeption

Ingolstadt, Deutschland | osama-altalmas.com | *Ab sofort verfügbar, bundesweit umzugsbereit* | *Führerschein B (voraussichtlich 01/2027)*

PROFIL

Robotics & Embedded Systems Engineer mit Schwerpunkt autonome mobile Roboter und Manipulatoren, der Roboterverhalten vom Algorithmus bis zum getesteten, sicherheitsbewussten System in ROS 2 mit C++ und Python bringt. Lager-AMR eigenständig entwickelt: eigene Nav2-Planer- und Controller-Plugins in C++, KI-gestütztes Docking auf 3–7 mm, YOLO-Hinderniserkennung und eine an ISO 3691-4 orientierte, per Fehlerinjektion verifizierte Sicherheitsschicht. Zudem UR5e-Sortierzelle mit eigener inverser Kinematik entwickelt und Forschung zur verbesserten RRT-Bahnplanung für Manipulatoren veröffentlicht. M.Sc. Robotik & KI (Jahrgangsbester); zwei Embedded-Praktika bringen praktische Erfahrung mit Firmware, Sensorik und realer Hardware.

ROBOTIK-PROJEKTE

Autonomer Lager-Tugger (AMR)

seit 03/2026

- AMR entwickelt: Routennetz, Andocken unter Ladungsträgern, Heben, Ausliefern; Fahrzeuglenkung, Hundegang, Drehen auf der Stelle.
- Eigene Nav2-Plugins in C++: RRT*-Planer mit Reeds-Shepp-Kurven, bidirektionaler Planer, Curvature-Tracking-Controller.
- Docking-Fehler von 3,0–6,6 mm (Ziel 10 mm / 1°) mit einem YOLO11-Bodenmarker-Modell (0 falsche Stationen bei 401 synthetischen Testmarkern).
- YOLO11 trainiert: Markerleser auf beschädigten AprilTags (abgenutzt, verdeckt), Hinderniserkennung auf Gazebo + realen Fotos.
- Fail-Closed-Sicherheit nach ISO 3691-4: Schutz-/Warnfelder, selbsthaltender Not-Halt, Stopp bei veralteten Sensordaten.

Robotische Sortierzelle mit Schutzzaun

2026

- UR5e-Sortierzelle am Förderband mit Pick-and-Place per Sauggreifer aufgebaut.
- OpenCV-HSV-Erkennung und Schnitt von Kamerastrahl und Bandebene liefern Boxposen im Weltkoordinatensystem.
- Eigene numerische IK (Least-Squares auf URDF-Kinematik) steuert den Arm.
- Türsensorik mit Pause, Reset und Not-Halt.

Verbesserte RRT-Bahnplanung – Masterarbeit

2024 – 2025

- RRT-Variante mit zielgerichtetem Sampling, optimierter Elternknotenwahl und Bézier-Glättung.
- 73,5 % weniger Rechenzeit und 67,9 % weniger Iterationen als RRT* über 50 Durchläufe.
- Simuliert an einem OpenManipulator-X-Modell.

BERUFSERFAHRUNG

Embedded Systems & IoT Engineer (Praktikum)

01/2026 – 02/2026

FreshNow · Dubai, VAE

- Embedded-/IoT-Abteilung des Unternehmens als erster Ingenieur aufgebaut; Berichte an Management und Investoren.
- Vierköpfiges Team (Mechanik, Software, Netzwerk) in zwei Monaten zu einem funktionsfähigen Smart-Tray-Prototyp geführt.
- ESP32-S3-Sensorsystem entwickelt: Wägezellen mit Abgleich über Endschalter, später 5 Laser-ToF-Sensoren an getrennten I²C-Bussen.
- Sensorbezogene Kalibrierung im Flash und Temperaturalarme.
- Wi-Fi-Protokoll und Heartbeat zwischen Gerät und Backend definiert.
- Verdrahtung, Löten, Tests und Installation übernommen; System-Platine in EasyEDA entworfen.

Embedded Hardware & Firmware Engineer (Praktikum)

09/2025 – 12/2025

Infinix Innovations · Dubai, VAE

- Vier interaktive Messe-Installationen mit Arduino/ESP32-S3-Firmware, Sensorik, Motoren und RFID entwickelt.
- Ein Arduino-Master koordinierte mehrere MCUs, angebunden an PCs über Serial und Ethernet-UDP.
- Encoder-gestütztes Drehrad in Betrieb genommen, das über Resolume Arena Inhalte auf einer LED-Videowand auslöst.
- Bereitschaftsingenieur vor Ort während der einwöchigen Messe.

AUSBILDUNG & PUBLIKATION

M.Sc. Robotiktechnologien & Künstliche Intelligenz

08/2023 – 05/2025

Bahçeşehir University, Istanbul, Türkei · Jahrgangsbester, Note 1,1

- Publikation zur Masterarbeit: Altalmas, O., & Bahceci, B. (2025). Obstacle Avoidance Path Planning for Industrial Robotic Manipulators Using Improved RRT with Bézier and Linear Interpolations. ICENSTED 2025, S. 523–534.

B.Sc. Mechatronikingenieurwesen

09/2019 – 07/2023

Bahçeşehir University, Istanbul, Türkei

- Bachelorarbeit: selbstbalancierender Roboter; Pendelmodell in Simscape, PID-Regelung am realen Roboter abgestimmt.

KENNTNISSE & SPRACHEN

Programmierung & Tools: C++ (17), Python, MATLAB/Simulink; Linux, Git/GitHub, GitHub Actions, colcon, pytest
ROS 2 & Navigation: ROS 2, Nav2-Plugin-Entwicklung, ros2_control, AMCL, SLAM, TF2, RViz, Gazebo Sim
Planung & Regelung: RRT / RRT*, Reeds-Shepp, Bahnverfolgung, Kinematik und IK, PID
Perzeption & KI: YOLO11 (Ultralytics) Detektion und Pose, OpenCV, synthetische Daten, GPU-Training (Kaggle/Colab)
Hardware: ESP32-S3, Sensorik und Encoder, SolidWorks
Sprachen: Arabisch (C2, Muttersprache), Englisch (C2), Türkisch (B2), Deutsch (A2, wird aktiv ausgebaut)