



# Nathan Corral

✉ [nathan.b.corral@gmail.com](mailto:nathan.b.corral@gmail.com)

🌐 <https://nathancorral.com>

🌐 [www.linkedin.com/in/nathan-corral](https://www.linkedin.com/in/nathan-corral)

☎ +49 160 9175 1918

👤 [NathanCorral](#)

Als Computer Engineer mit einem Master-Schwerpunkt in Computer Vision und Robotik bin ich motiviert, die angewandte Automatisierung durch modernste Deep-Learning-Lösungen voranzutreiben.

## Berufserfahrung

### ■ Humanoid Robots Lab – Universität Bonn

09.2021 – 09.2022

*Wissenschaftlicher Mitarbeiter*

Bonn, Deutschland

- Mitgewirkt an Forschung und Veröffentlichungen in Bereich "Personalized Robot Navigation".
- Programmierung der ROS-Schnittstelle für die 3D-Lokalisierung von Menschen mit einer RGBD-Kamera unter Nutzung von Deep Learning und Implementierung dieser Funktion auf einem realen Roboter zur autonomen Navigation.
- Verwendung des fotorealistischen Simulators iGibson (PyBullet-Backend) zur Generierung von Daten für einen Deep-Reinforcement-Learning-basierten Path Planning Algorithm.
- Aufbau und Durchführung einer Nutzerstudie zur Bewertung der Mensch-Roboter-Interaktion mit einem VR-Headset und anschließender Umsetzung auf realer Roboterhardware.

### ■ Head Rush Technologies

12.2019 – 04.2020

*Vertragsingenieur*

Boulder, USA

- Vertragsarbeit zur Entwicklung der Firmware auf einem ATmega328PB-Mikrochip für ein Proof-of-Concept-System.
- Arbeit umfasste die Programmierung eines durch Interrupts ausgelösten Zahnrad-Sensors, RS485-Kommunikation, einer PWM-gesteuerten Bremse sowie Logik für endliche Zustandsautomaten.
- Durchführung von Feldtests und Erstellung der Projektdokumentation.
- Der Erfolg dieses Prototyps führte zu einer weiteren Entwicklung, die letztendlich als ihre „Catch-and-Hold-Technology“ veröffentlicht wurde.

### ■ Aqronos

11.2018 – 12.2019

*Softwareentwickler*

Denver, USA

- Entwicklung von ROS-Nodes zur Visualisierung des LiDAR-Prototyps des Unternehmens.
- Strukturierung von UDP-Paketen und Programmierung beider Seiten der Sende- und Empfangsmodule.
- Interaktion mit einer REST-API auf dem eingebetteten System zur Konfiguration von Hyperparametern.
- Filterung von Punktwolken und Gruppierung von Objekten mit der C++ Point Cloud Library.

### ■ Creative Edge LLC

08.2017 – 09.2018

*Softwareentwickler*

Denver, USA

- Entwicklung von Anwendungen für das Kryptowährungs-Mining unter Windows und Linux.
- Erstellung von Software zur Verwaltung von Betriebssystemtreibern, Systemkonfigurationen und Tools von Drittanbietern.

## Bildung

■ M.Sc. Universität Bonn  
10.2020 – 09.2023

*Informatik*  
Note: 1.7

■ B.Sc. University of Illinois Urbana-Champaign  
08.2013 – 05.2017

*Computer Engineering*  
GPA: 3.55/4.0

## Masterarbeit

2023 ■ **Stochastic Transformer for Prediction of Multiple Futures**

Diese Arbeit baut auf den Grundlagen der Stochastic Video Generation<sup>1</sup> und Variational Transformers<sup>2</sup> auf und erweitert deren Anwendungen zu einem vielseitigen, aufgabenunabhängigen, stochastischen Vorhersagenetzwerk. Diese Arbeit trug Folgendes bei:

- Eine neuartige, transformerbasierte Prädiktorarchitektur, die eine Verteilung über mögliche Zukünfte lernen kann.
- Detaillierter Vergleich mit anderen stochastischen Modellen in der Videovorhersage, mit einer höheren Structural Similarity in frameweisen Vergleichen.
- Anwendung im Bereich Human Pose Prediction, mit der Generierung von über 8 Sekunden fortgesetztem Gehen nach den ersten 0,3 Sekunden Seed-Bewegung.

## Projekte

2024 ■ **ROS 2 Whisper**

[Video](#), [Source](#)

Als Erweiterung dieses Open-Source-Projekts habe ich eine grenzenlose, Live-Audiotranskription implementiert. Geschrieben in C++ legt mein Codebeitrag den Fokus auf:

- Skalierbarkeit: Through using both inheritance and composition in object-oriented programming behavior.
- Effizienz: Through intentional memory management, thread-safe callbacks and work splitting across multiple nodes.
- Einfachheit in der durchdachten Implementierung complex merging algorithms.

■ **ROS 2 Computer Vision**

[Video](#), [Source](#)

Das Ausführen mehrerer Computer-Vision-Modelle (DETR, Maskformer), die für verschiedene Datensätze/Aufgaben trainiert wurden, auf einem Live-Kamerastream stellt mehrere Implementierungsherausforderungen dar. Dieses Python-Repository bietet eine Lösung für:

- Das Herunterladen und Ausführen von State-of-the-Art-Modellen aus Hugging Face als asynchrone ROS 2 Nodes.
- Das Hosten eines Label Server, um Modell-Ausgaben in einer globalen Datenbank neu zu adressieren.
- Die Anzeige von Segmentation Masks und Bounding Boxes als Matplotlib Animations.
- Das Veröffentlichen von Datensatzbildern zur wiederholbaren Evaluierung von CV-Modellen.

<sup>1</sup>Denton et al., "Stochastic video generation with a learned prior." ICML 2018

<sup>2</sup>Lin et al., "Variational transformers for diverse response generation." arXiv 2020

## Projekte (continued)

### ■ Semantic Search using Facebook AI Similarity (FAISS)

[Source](#)

Dieses Projekt implementiert die ersten Schritte der Retrieval-Augmented Generation (RAG) (endet vor „Generation“). Ich führe Web-Scraping, Datensatz-/Abfrage-Einbettung und Ähnlichkeitsbewertung durch, um Dateneinträge basierend auf einer Abfrage in natürlicher Sprache abzurufen.

## Veröffentlichungen

- J. de Heuvel, N. Corral, et al. “Learning depth vision-based personalized robot navigation from dynamic demonstrations in virtual reality” *IROS*, 2023

## Fähigkeiten

Languages	■	· English (Native) · Deutsch (C1)
Stärken	■	· Problemlösung · Zusammenarbeit · Zuverlässig · Technische Dokumentation · Fleißig
Coding	■	· C++ · Python · Bash · C · LaTeX · Java
Software	■	· Linux/Ubuntu · GitHub · Docker · ROS/ROS2 · QEMU · Hyperstack · AWS EC2
Libraries (C++)	■	· std · chrono · Point Cloud Library · nlohmann/json · curl
Libraries (Py)	■	· PyTorch · Hugging Face · TensorFlow · Matplotlib · Pandas · OpenCV · NumPy · scikit-learn
Wissen	■	· Agile · REST API · Test-driven Development · POSIX · Object Oriented Programming · Data Structures
Robotics	■	· Forward/Inverse Kinematics · SLAM · Path Planning · PID / Model Predictive Controllers · Kalman (Bayes) Filters
Deep Learning	■	· Computer Vision · Generative AI · Large Language Models · Gradient Descent Optimization · Retrieval-Augmented Generation · Reinforcement Learning · Point Cloud Processing · CUDA
Simulators	■	· CARLA · iGibson · (Py)Bullet · Gazebo · Webots
Microcontrollers	■	· UART/I2C/SPI · Systems on Chip · Real-Time Systems · Interrupt Triggers · Discrete Signal Processing