

Roman Knyazhitskiy

✉ cv@knyaz.tech — 🌐 knyaz.tech — 💬 knyazer

Summary

MPhil student in Machine Learning at Cambridge with extensive hands-on experience in robotics, autonomous systems, and deep learning. Led RoboCup team developing real-time control systems with Model Predictive Control and differentiable simulation in JAX. Proficient in Python, C++, PyTorch, JAX and Rust. Research background includes deep neural networks for path forecasting and planning, computer vision for robotic applications, and reinforcement learning. Multiple award-winner in hackathons and robotics competitions.

Education

MPhil in MLMI (Machine Learning), University of Cambridge 10/2025 - 09/2026

BSc Computer Science and Engineering, TU Delft 09/2022 - 07/2025

- GPA: 8.7/10. Cum Laude + Honours.

Work Experience

Machine Learning Engineer, Delft Mercurians (RoboCup Team) 05/2023 - 09/2025

- Led team of 2-5 engineers developing AI control systems for autonomous soccer robots competing in RoboCup
- Designed and implemented Model Predictive Control system for real-time trajectory optimization with dynamic obstacle avoidance
- Built continuous-time differentiable physics simulator using JAX+Equinox for robotic dynamics modeling
- Developed PyQt interface for Kalman filter calibration with protobuf data visualization for better sensor fusion
- Integrated Python ML models into Rust codebase with tokio async threading and inter-process communication

Research Associate, TU Delft 03/2023 - 08/2025

- Researched deep learning applications including LLM-based code analysis and Prior-Data Fitted Networks for AutoML
- Delivered technical talks on optimization methods including higher-order optimizers and spectral characteristics
- Investigated MCMC acceleration methods and participated in departmental research on diffusion models

Applied Machine Learning Intern, Central Robotics Institute 06/2021 - 07/2021

- Developed computer vision system for robotic drawing applications with image-to-line conversion
- Implemented custom image segmentation and space-filling curve algorithms to maximize prettiness of the output

Publications

- [1] J. Luijmes, A. Gielisse, R. Knyazhitskiy, and J. van Gemert. ARC: Anchored representation clouds for high-resolution INR classification. In *ICLR 2025 Workshop on Weight Space Learning*, 2025. Accepted.

- [2] R. Knyazhitskiy and P. R. Van der Vaart. A simple scaling model for bootstrapped DQN. 2025. Under review.

Selected Projects

Bootstrapped DQN Scaling Laws, JAX, Deep Reinforcement Learning 2025

- Conducted large-scale empirical study (~40,000 configurations) on ensemble-based exploration in deep RL
- Implemented Bootstrapped DQN with multi-GPU distributed training and automatic device sharding
- Achieved 95% GPU utilization by using compressed state representation to minimize memory transfers

Lyapunov-Stabilized RL for Continuous Control, Python, JAX, Brax Physics 2024

- Implemented actor-critic framework for physics-based locomotion with truncated backpropagation through time
- Developed gradient stabilization using Lyapunov multipliers for long-horizon sequential decision-making
- Built parallel training infrastructure with 40 simultaneous environments and multi-host distribution
- Tested on Brax Ant locomotion achieving stable policy learning in complex continuous control tasks

Stack-Associated Beam Tracing Renderer, C++20, Computer Graphics 2022

- Built 3D rendering engine based on sparse voxel octree traversal with custom optimization techniques
- Implemented Gilbert-Johnson-Keerthi algorithm for convex collision detection from scratch
- Developed hierarchical spatial partitioning with distance-based traversal and adaptive LOD
- Created custom linear algebra library for vectors, matrices, transformations, and geometric primitives

Nano JAX GPT with Flash Attention, JAX, Equinox, Deep Learning 2023

- Implemented GPT transformer from scratch with custom Flash Attention using JAX Pallas for GPU/TPU acceleration
- Supported distributed multi-device training with sharding and mixed-precision (bfloat16/float32)
- Explored speculative decoding with early-exit prediction heads for 30-50% inference speedup
- Looked into using custom multi-step optimizer, multi-token prediction and other ideas

Silver-qt: Sign Language Recognition, Python, PyQt5, ONNX, Computer Vision 2019

- Developed real-time sign language recognition with three-stage neural pipeline: YOLOv5 hand detection, autoencoder feature compression, LSTM sequence classification across 51+ sign classes
- Implemented multi-threaded processing with QThreadPool for concurrent video capture and inference
- Built comprehensive preprocessing pipeline with letterbox normalization, NMS with configurable IoU thresholds, and center-crop transformations
- Won Silver Medal at AIIJC International AI Competition for Juniors

IEFT-PFN: Hyperparameter Optimization with Transformers, JAX, Equinox, Bayesian Optimization 2025

- Built transformer-based system for hyperparameter optimization using in-context learning to predict training curves

- Implemented 8-layer causal transformer with histogram-based probabilistic decoder for uncertainty quantification
- Developed three spatial weighting schemes: covariance-based, learned attention, and Euclidean distance
- Achieved 15-20% improvement over baseline with learned attention on real benchmark datasets

Honours and Awards

- **Best Software Solution**, RoboCup World Championships, Sydney (2019)
- **1st Place**, RoboCup Junior National Competitions (2016, 2017, 2019)
- **1st Place**, Bunq Hackathon 6 (2025) – €30,000 prize
- **2nd Place & Special Prize**, Epoch AI Hackathon (2024)
- **Silver Medal**, AIIJC International AI Competition for Juniors
- Winner of numerous robotics, competitive programming, astrophysics, and mathematics competitions

Technical Skills

- **Programming:** Python, C++, C, Rust, CUDA (familiar)
- **Deep Learning:** PyTorch, JAX/Equinox, TensorFlow/Keras, ONNX Runtime
- **Systems:** Linux, Nix, Docker, Multi-threading, Distributed Computing, Version Control (Git)

Open Source Contributions

- Enhanced [jaxtyping](#) and [Equinox](#) ML libraries, enabling runtime typechecking for Jupyter notebooks
- Contributed to [Gymnax](#) JAX RL environments (800+ GitHub stars)
- Fixed critical infinite loop bug in libccd collision detection library (C++), used by thousands of people