Luca Brembilla

luca.brembilla@polimi.it | GitHub | LinkedIn

PhD researcher at the AIDA group, Politecnico di Milano (DEIB), working on temporal self-supervised learning for perception in autonomous driving, with multimodal (camera+LiDAR) representation learning as the longer-term direction (advisors: M. Corno, G. Boracchi). MSc graduate, 110/110 cum laude, Computer Science & Engineering (AI specialization), Politecnico di Milano.

EXPERIENCE

PhD Researcher - AIDA, Politecnico di Milano

[02/2026 - Present]

What Does the Future Buy? Separating projection from prediction in LiDAR pre-training (ELLIS Summer School 2026, poster). Future-predictive pre-training (**TREND**) changes two things at once: it adds a recurrent, ego-action-conditioned transformation between encoder and renderer, and it adds future supervision targets. Since only the encoder reaches the detector, I built a controlled ablation platform that varies the two one at a time (Direct / Projection / Recurrent Projection / Forecasting / No anchor). On nuScenes few-shot (TransFusion-L), detection separates *the transformation, not the future*: every present-anchored transformation arm beats Direct by ~3x the seed noise, while Forecasting, Recurrent Projection and a plain action-free Projection sit within a 0.45 mAP band; the best arm is the present-only projector, at **~47% less pre-training compute per epoch**. Future supervision has a cost: car velocity error grows **~15%** monotonically with pre-training dose. Cross-checked on ONCE (CenterPoint) against a baseline trained to convergence.

DriveRT (personal project, open source, in progress). A compiler and runtime for sparse, dynamic autonomous-driving models: captures unmodified PyTorch code (voxelization, sparse convolution, BEV, decoding, NMS) into one canonical graph and compiles it into a single `.drt` file run by one small C library, with no Python/PyTorch at runtime. Whole-program optimizations: dataflow-proven row-selective head evaluation, shared neighbour maps, device-side counts, per-GPU tuned sparse-convolution kernels. **VoxelNeXt runs end to end 1.85x faster** than OpenPCDet+spconv with identical boxes; 851 tests.

Thesis co-supervision (3 MSc students): (i) self-supervised pretraining for the UniTR multimodal fusion transformer (iBOT/DINOv2-style masked self-distillation; diagnosed a positional-shortcut representation collapse in the LiDAR branch); (ii) 3D object detection and semantic segmentation with OpenPCDet; (iii) synthetic data generation with NVIDIA Cosmos.

Production perception (NiuLinx / AIDA platform). Trained and benchmarked multimodal camera-LiDAR 3D detectors, including models with an attention-based LiDAR backbone, on nuScenes and in-house data. TensorRT deployment with custom plugins (BEVPool, etc.) and INT8 quantization for real-time inference on target hardware. Migrated the ADE Docker environment from **TensorRT 8 to 10** and **CUDA 11.8 to 13**, fixing the full library and compilation stack.

Tecnológico de Monterrey collaboration. Following the project deploying the AIDA perception stack on an autonomous minibus in Monterrey, Mexico (on-site demos, Feb 2026).

Public-road testing. Authorized by the Italian Ministry of Infrastructure and Transport (MIT) as a **test supervisor for autonomous-vehicle experimentation on public roads** (D.M. 70/2018, art. 9; 2026).

> *Perception software authored at Polimi was part of the technology transfer to NiuLinx, a €38M-funded autonomous-driving spin-off from Politecnico di Milano (2026).*

Research Engineer - AIDA, Politecnico di Milano

[05/2025 - 02/2026]

Contributed to the Perception team of the AIDA project, the only Italian platform certified by the Ministry of Public Transport for public road testing.

- Trained and benchmarked modern 3D detectors (baseline: PointPillars).
- Deployed transformer-based model in the ROS 2-based stack via ONNX/TensorRT, achieving 30 ms inference on target hardware.
- Implemented custom CUDA kernels to enhance the stack, e.g. parallelize PointCloud2 unpacking on GPU, reducing preprocessing latency about 6x (9 ms to 1.5 ms).
- Built a **late-fusion LiDAR-camera detection** with an occlusion-aware cross-modal module (geometric proposals vs. visual features) and DIoU matching to suppress false positives.
- Led a **Strategy-pattern** refactor decoupling LiDAR detectors for flexible model integration; managed the project Docker environment (`pcdet`, `spconv`, specialized libraries).

EDUCATION

PhD in Information Engineering - Politecnico di Milano, DEIB [02/2026 - Present]
Research: self-supervised learning for perception in autonomous driving.
Advisor: Prof. Matteo Corno | Co-advisor: Prof. Giacomo Boracchi.

Master's Degree in Computer Science and Engineering (Artificial Intelligence specialization)
Politecnico di Milano [09/2022 - 04/2025] - *Final grade: 110/110 cum laude*

Bachelor's Degree in Computer Engineering
University of Bergamo [08/2019 - 02/2022] - *Final grade: 105/110*

PUBLICATIONS

S. Mentasti, **L. Brembilla**, and M. Matteucci, "*Real-Time LiDAR Object Detection on Edge Devices via Adaptive Cropping*," International Conference on Image Analysis and Processing (ICIAP), 2025. **Oral presentation.**

MASTER'S THESIS

Real-Time LiDAR Object Detection on Edge Devices via Adaptive Cropping - AIRLab, Politecnico di Milano
Adaptive cropping strategy: **50%** point-cloud reduction with minimal mAP loss; **2x FPS** on edge devices. Published at ICIAP 2025 (oral).

SKILLS

- **Languages:** Italian (native), English (C1)

- **AI / ML:** PyTorch, TensorFlow, deep learning, perception, self-supervised learning
- **Inference & Deployment:** TensorRT (INT8, custom plugins), ONNX, CUDA kernels
- **Autonomous Driving:** ROS 2, Multimodal fusion, BEV perception, 3D object detection, segmentation
- **Infrastructure:** Docker (ADE), Linux, NVIDIA CUDA ecosystem, nuScenes
- **Compilers & Systems:** graph capture (PyTorch dispatcher), IR design, sparse convolution kernels
- **Programming:** Python, C++, C, CUDA; JavaScript, Java