Niccolò **Ferrari**

Senior Machine Learning Engineer & Researcher | Ph.D.

About me

Senior Machine Learning Engineer with a Ph.D. research background focused on industrial computer vision for visual inspection, anomaly detection, segmentation, and production deployment. I design deep learning models and C++/Python inference software for real-time inspection systems. My work sits between applied research and production software, from model design and experiments to machine integration and edge or on-prem deployment. I have published work and open-source research code for GRD-Net, PatchCore-style methods, and graph memory transformers.

Personal

Niccolò Ferrari
Cormano (MI), Italy
08/02/1993

Areas of specialization

Industrial Computer Vision • Visual Inspection • Anomaly Detection • Segmentation • Production ML • C++ Inference

Interests

Artificial intelligence and deep learning research • Game development • Electronic circuit design • PCB design • SMD soldering • 3D printing • Climbing • Bouldering • Trekking • Latin • Books • Films • Cacti cultivation

Main languages

`C++ / Python / Java / C`

Main frameworks

`PyTorch / TensorFlow / Halcon / OpenCV / scikit-learn`

Professional Experience

Dec 2024–Now — **Senior Machine Learning Engineer & Researcher**
NAIS S.R.L. · Bologna, Italy
Design and deploy production ML and real-time computer vision systems for edge devices. Built and deployed a C++20 embedded vision system on low-resource edge hardware for food-line counting, classification, measurement, and statistical data collection, achieving 99.99% reliability. Lead an industrial ML project on vibration-based mechanical data. Serve as Technical Lead of the AI team, coordinating model deployment, applied research, and technical choices for project feasibility.

Apr 2018–Dec 2024 — **Machine Learning Engineer**
BONFIGLIOLI ENGINEERING · Ferrara, Italy
Designed and deployed ML and computer vision for pharmaceutical visual inspection, using Halcon for algorithmic machine vision and TensorFlow/PyTorch for deep learning on rare-defect cases and larger industrial datasets. Built a C++ inference runtime for on-board clusters; achieved >95% OOD anomaly-detection accuracy in an industrial case. Managed Linux training/inference servers with high-end NVIDIA GPUs, including two A100-class machines, for industrial ML experimentation and deployment support.

Nov 2021–Nov 2024 — **Ph.D. in Deep Learning and Computer Vision**
UNIVERSITY OF FERRARA · Ferrara, Italy
Research in deep learning and computer vision for OOD anomaly detection and segmentation. Designed two Python-based architectures; one was deployed through a C++17 inference pipeline for pharmaceutical inspection, resulting in two publications.

Sep 2017–Mar 2018 — **Curricular Internship**
BONFIGLIOLI ENGINEERING · Ferrara, Italy
HMI Java developer; backend and frontend with Industry 4.0 SPC toolchain for line workflow control.

May 2015–Oct 2015 — **Curricular Internship**
CIAS · Ferrara, Italy
JSP web service developer; deployed a hospital questionnaire system to track healthcare-associated infections.

Degrees

Oct 2025–Now — **Game Programming + AI Programming**
LEARNING PATH · Digital Bros Game Academy
Learning Path Master Online Game Programming + AI Programming

Nov 2021–Apr 2025 — **Computer Science Engineering - Deep Learning**
PH.D. · University of Ferrara
EQF 8–Deep Learning architectures for Computer Vision applications

Oct 2015–Mar 2018 — **Computer Science and Automation Engineering**
MASTER · University of Ferrara
EQF 7–110/110 cum laude et encomium

Oct 2012–Nov 2015 — **Computer Science and Electronic Engineering**
BACHELOR · University of Ferrara
EQF 6–110/110 cum laude

Programming

C++	Deep learning inference runtimes, C++ backends, HMI systems, and edge/on-prem deployment
Python	Deep learning research, training pipelines, tooling, and production ML workflows
Halcon script	Computer vision algorithms deployed on pharmaceutical inspection machines
Java	HMI frontends and Industry 4.0 / SCADA plugins
C	Systems and embedded programming
Bash	Linux automation, scripting, and development tooling
R	Industry 4.0 tooling and data-oriented libraries

Operating Systems

Linux	Daily development environment; on-prem GPU training and inference infrastructure
FreeBSD	Applied development on FreeBSD systems
Windows	General desktop environment

niccolo.ferrari.93@gmail.com
niccolo.ferrari@unife.it
+39 366 270 5735
+39 388 821 1473
NickF93
niccolo-ferrari-93bo
pigreco.xyz
0000-0002-2578-3211

Languages Italian native
English B2
License Driving License B (E.U.)

Publications and Research Outputs

2023	Niccolò Ferrari, Michele Fraccaroli, Evelina Lamma, "GRD-Net: Generative-Reconstructive-Discriminative Anomaly Detection with Region of Interest Attention Module", International Journal of Intelligent Systems, 2023. https://doi.org/10.1155/2023/7773481
2024	Niccolò Ferrari, Nicola Zanarini, Michele Fraccaroli, Alice Bizzarri, Evelina Lamma, "Integration of deep generative Anomaly Detection algorithm in high-speed industrial line" [Under review, submitted to Springer Nature]. https://dx.doi.org/10.2139/ssrn.4858664
2025	Ph.D. thesis: *Machine Learning Techniques for Anomaly Detection in Pharmaceutical Quality Control*, University of Ferrara. https://hdl.handle.net/11392/2587174
2026	Niccolò Ferrari, Oligert Osmani, Evelina Lamma, "Mahalanobis PatchCore: Covariance-Aware and Streaming-Compatible Industrial Anomaly Detection" [Under review, submitted to EAAI], arXiv:2605.27748 [cs.CV], 2026. https://doi.org/10.48550/arXiv.2605.27748
2026	Nicola Zanarini, Niccolò Ferrari, Evelina Lamma, "Graph Memory Transformer (GMT)", arXiv:2604.23862 [cs.LG], 2026. https://doi.org/10.48550/arXiv.2604.23862

Research Supervision

M.Sc. thesis	*Improving PatchCore for Visual Anomaly Detection in Pharmaceutical Product Inspection*. Author: Oligert Osmani; supervisor: Evelina Lamma; co-supervisors: Niccolò Ferrari and Davide Luisari.

Certificates and Courses

Deep Learning & ML	TensorFlow Advanced Techniques; TensorFlow 2.0 Practical Advanced; Diffusion Models; Quantization Fundamentals and Quantization in Depth; Physics-Informed Neural Networks (PINNs); Graph Neural Networks (GNN)
C++ / Systems / DevOps	Modern C++ (C++11/14/17); Pure C++20; Data-Oriented C++; ARM CMSIS-RTOS; Git/GitHub; Jenkins CI/CD
Game Development	Epic Games Game Design; Unreal Engine Fundamentals; Blueprint Scripting
Local LLMs	Harnessing Ollama local LLMs with Python

Frameworks, Ecosystems and Tools

PyTorch / TensorFlow	Model training, research prototyping, and deployment workflows for academic and industrial work
Halcon / OpenCV	Halcon-based algorithmic inspection work; OpenCV used in later computer vision prototyping and deployment
SQL (MySQL & PostgreSQL)	Production data stores and training-data pipelines
Git / GitHub / GitLab	Branching, code review, collaborative repository management, and release-oriented workflows
Jenkins	CI jobs, build automation, and testing/deployment support
ONNX / OpenVINO / TensorRT	Model export, inference optimization, and deployment on edge and GPU-accelerated targets
Docker	Reproducible ML environments and deployment support
scikit-learn / NumPy / Pandas	Data processing, ML utilities, experiments, analysis, and feature preparation
MLflow / TensorBoard	Experiment tracking and training diagnostics
AI-assisted coding	Codex and Claude Code for implementation, refactoring, tests, and review, with architecture and validation remaining my responsibility

ML Technologies and Skills

Visual anomaly detection	OOD anomaly detection, visual inspection, segmentation, GANomaly, PatchCore, GRD-Net, autoencoders and VAE, and knowledge distillation in research and production contexts
Deep generative models	GANs, variational autoencoders, normalizing flows, nflows, and diffusion models
Computer vision architectures	CNNs, ResNet, ViT, and YOLO models for research and edge/on-prem vision projects
Transformers and LLMs	Transformer architectures and local LLM workflows for research experiments
Production ML deployment	Edge AI, embedded inference, real-time computer vision, and C++/Python inference pipelines

Research Software

GRD-Net	Official implementation and industrial high-speed line integration for deep generative anomaly detection. https://github.com/NickF93/GRD-Net
MH-PatchCore	PatchCore/Mahalanobis PatchCore code for submitted EAAI paper. https://github.com/NickF93/MH-PatchCore; tiny-edge experiments: https://github.com/NickF93/mh-patchcore-tinyedge
GMT	Co-authored Graph Memory Transformer source code. https://github.com/Nemesis533/GMT-GraphMemoryTransformer
AgentOrchestrator	Control-plane software for coordinating coding agents in agent-assisted development workflows. https://github.com/NickF93/AgentOrchestrator

Professional Skills

Industrial CV & deployment	Build deep learning systems for visual inspection, anomaly detection, and segmentation, then move them toward real-time edge or on-prem inference with C++/Python, ONNX, OpenVINO, and TensorRT
Vision software	Work with Halcon for pharmaceutical machine-vision algorithms and PyTorch/TensorFlow for deep learning, including PatchCore-style methods and GRD-Net research software
Production engineering	Translate machine and project requirements into maintainable C++/Python/Java software, backend services, SQL data flows, tests, and Linux deployment scripts
Leadership & delivery	Coordinate feasibility and implementation choices for the AI team. Use Git/GitHub/GitLab, Docker, Conda, Jenkins, MLflow, TensorBoard, PyTest, Codex, and Claude Code for review, testing, tracking, and reproducible delivery