



Alexandru Ont

📍 **Home:** Strada Principala nr. 17, 437276, Mogosesti, Romania

✉ **Email address:** alex.ont@icloud.com ☎ **Phone:** (+40) 741821917

🌐 **LinkedIn:** <https://www.linkedin.com/in/alexandru-mihai-ont-40b8a833a/>

Date of birth: 25/07/2002 **Place of birth:** Baia Mare, Romania **Nationality:** Romanian

ABOUT MYSELF

Machine Learning Engineer and Computer Science student at UTCN CUNBM with a strong passion for building practical AI systems. Experienced in developing and deploying machine learning models, particularly in computer vision applications. Continuously motivated to learn new technologies and improve my skills while creating impactful software solutions.

WORK EXPERIENCE

Ensemble Software

City: Baia Mare | **Country:** Romania

[19/08/2025 - Current]

ML Engineer

- Develop and deploy computer vision models for real-world applications, including object detection systems using YOLO architectures.
- Design and implement end-to-end ML pipelines from data preparation and model training to production deployment.
- Build scalable APIs and services in Python for integrating AI models into production systems.
- Optimize inference pipelines and deploy models on edge devices and specialized hardware accelerators.
- Develop AI-assisted tools using modern ML frameworks and integrate monitoring solutions using Docker, Loki, and Grafana.

PROJECTS

[01/03/2025 - Current]

License Plate Object Detection AI Model

In this project I built and trained a complete object detection system using **YOLOv8** and **OpenCV**, capable of detecting license plates in real time from video files or a live webcam feed.

What I Did

- I **trained a custom YOLOv8 model** on a labeled dataset for **700 epochs**, optimizing it for accurate license plate detection.
- I developed the entire detection pipeline in Python, integrating the trained model to process video streams using **OpenCV**.
- I implemented:
 - Real-time detection with bounding boxes filtered by confidence score.
 - Frame annotation and automatic video saving using VideoWriter.
 - Webcam and video file compatibility for flexible input.
- I designed the code to be modular and adaptable, allowing future improvements or extensions.

Project Impact

- I successfully delivered a **self-contained, real-time object detection system** that can be used for license plate recognition.
- Built a full ML pipeline from data to real-world use, integrating the trained YOLOv8 model into the AutoRos robot control system for real-time license plate detection.

Link: <https://github.com/alexandruont/Object-Detection-Model>

[15/02/2025 - Current] **Learnify**

In this project I built a Learning Management System, mainly focused on topics related to Computer Science where instructors can create courses and quizzes, and students can enroll, study materials (PDFs), take quizzes, and track results.

Key Highlights:

- Implemented user authentication with role-based dashboards (Admin, Instructor, Student).
- Built full **CRUD** operations for courses, quizzes, and question management.
- Integrated PDF-based course delivery with quiz linkage and result tracking.
- Designed a responsive UI using Bootstrap and JavaScript.
- Developed a normalized relational database with foreign keys and cascading rules.

Tech Stack:

PHP, MySQL, HTML/CSS, Bootstrap 5, JavaScript, XAMPP

Link: <https://github.com/alexandruont/Learnify>

[01/03/2025 - Current] **AutoRos Project**

As part of the **AutoRosPrct** project, I was responsible for designing and developing both the **frontend interface** and the **backend logic** that enabled real-time control and monitoring of a physical robot. My work focused on integrating web technologies with hardware-level communication protocols.

- I created a **responsive web UI** using HTML, CSS, and JavaScript that allows users to control robot movement and its 6-axis robotic arm.
- Implemented **range sliders** and **button controls** with real-time feedback using JavaScript and DOM manipulation.
- Integrated **Socket.io** for real-time, bidirectional communication between the browser and server.
- Developed a **Node.js server** using Express to serve static content and manage WebSocket connections.
- Established a **TCP client** inside the server to communicate with the robot hardware using structured JSON commands.
- Simulated a **live camera feed** by rotating a series of images to give a dynamic user experience.

How I Helped the Project

- My work enabled the robot to be **controlled remotely and interactively** through a web browser, increasing usability and accessibility.
- Reduced latency and improved responsiveness through optimized WebSocket handling and modular code structure.
- Built a system that can be easily extended to control **multiple robots** or receive **telemetry data** forming the foundation for future development.

Link: <https://github.com/alexandruont/AutoRosPrct>

[30/05/2025 - 02/06/2025] **Programming Competition Finalist**

Developed a graph traversal algorithm using **Depth-First Search**, enhanced with **pruning (Branch & Bound)** and **heuristic sorting** to identify the highest-scoring cyclic path in a weighted graph under time and visit constraints.

- Implemented artifact tracking, cycle detection, and time-limited execution for real-world efficiency
- Optimized search space via upper-bound estimations and greedy neighbor selection

- Used structured input/output handling with CSV and TXT files
- Strengthened problem-solving, algorithm design, and performance tuning skills

Link: <https://github.com/alexandruont/ProgrammingCompetition2025>

EDUCATION & TRAINING

[02/10/2023 - Current]

Computer Science Student

UTCN Centru Universitar Nord Baia Mare stiinte.utcluj.ro

[01/09/2017 - 30/07/2020]

DIPLÔME DU BACCALAUREAT GENERAL – Scientific Series, European Section (English)

Lycee Isaac Newton <https://lyceenewton.fr/>

LANGUAGE SKILLS

Mother tongue(s): Romanian | French

Other language(s):

English

LISTENING: C1 **READING:** C1 **WRITING:** C2

SPOKEN PRODUCTION: C1 **SPOKEN INTERACTION:** C1

SKILLS

Programming

Python | JavaScript | SQL | Bash

Machine Learning & AI

Machine Learning (AI, DL, CNN) | Computer Vision | Data Preprocessing | Model Training & Evaluation | Transformers | OCR | RAG | Agentic AI

Frameworks & Libraries

PyTorch | OpenCV | NumPy | Pandas | Tensorflow | ONNX

Deployment & Infrastructure

Docker | Edge AI | Linux | AWS (S3, Lambda) | NVIDIA CUDA | REST APIs (FastAPI / Flask)

Monitoring & Tools

Git | Grafana | Loki | MLFlow