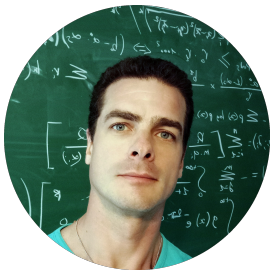

Emmanuel **ROCHE**

Senior Software Engineer · System Admin · 3D expert · Data Scientist · Blockchain enthousiast · AI Advocate

06130 Grasse, FRANCE

(+33) 620315649 · roche.emmanuel@gmail.com · nervtech.org

roche-emmanuel | emmanuel-roche | @magik_engineer | @magik_engineer

French: **Native** | English: **Fluent**

Applying to position: Graphics Software Engineer, GPU Driver - Reality Labs

Work Experience

GMV · Senior Software Engineer · Nov. 2015 - Present

May 2023 - Jun. 2023
2 months
Client: EUMETSAT

MTG-LI Off-Line Tools - Stage 3 (RESPASE Task 10)

Service Level Manager / Technical Lead Engineer

- Oversaw repository maintenance, milestone tracking, issue management, and review of merge requests from colleagues.
- Implemented support for Micro Vibration Windows (MVW) handling in LI-PVT GUI application.
- Added support for visualization and browsing of the micro-vibration window (MVW) grid.
- Developed a configuration mechanism within the GUI to define desired target micro-vibration windows for each optical channel.
- Designed and implemented a Python script to generate dummy data for micro-vibration window measurements.
- Integrated functionality for loading and visualizing existing micro-vibration window measurements.
- Introduced additional minor features, including status bar updates to enhance the user experience.
- Conducted thorough testing of the tool, ensuring its stability and reliability in various scenarios.

`python` · `pyQt5` · `vispy` · `xarray` · `yaml` · `satpy` · `pytest` · `pandas` · `netcdf`

Mar. 2023 - Jun. 2023
4 months
Client: EUMETSAT

FCI-ATT v2.0 Implementation (RESPASE Task 9)

Service Level Manager / Technical Lead Engineer

- Managed the project repository, handling issues, milestones, and merge requests.
- Extended the FCI ATT application and improved cloud masks generated by the spectral tests.
- Implemented additional spectral tests, including complex computations and comparisons.
- Developed support for loading and pre-processing external reference data for comparison with computed results.
- Implemented computation of validation statistics and metrics, as well as visualization of data set comparisons.
- Introduced functionality for modifying and combining cloud masks, supporting binary and floating-point data sets.
- Updated project documentation, incorporating new elements and components in RsT.
- Worked on the animation of L1C data, allowing users to navigate and observe cloud movement.
- Ensured the application produced valid results through rigorous unit testing.
- Developed using Visual Studio Code on both Windows and Linux platforms.
- Employed GPU-based unit testing to compare results between GPU and CPU computations.

`python` · `jupyter` · `GLSL` · `OpenGL ES2` · `vispy` · `satpy` · `multithreading` · `Sphinx` · `Visual Studio Code`

Oct. 2021 - Jun. 2023
1 year, 9 months
Client: EUMETSAT

SCA/Ascat Deblurring Study

ASCAT Support Engineer

- Collaborated with GMV Poland, IsardSAT, and GMV France to evaluate ASCAT and SCA imagery deblurring algorithms.
- Led Python implementation phase using Jupyter notebooks and Numba for algorithm optimization.
- Conducted reverse computations for comparing deblurred results with sharp images using Perlin noise and high-res grid computation.
- Utilized NumPy, X-Array, Matplotlib, and SciPy for NetCDF file handling, result visualization, and SVD decomposition on large matrices.
- Developed gradient descent algorithm with rmsprop, adamax, nadam, etc., incorporating various descent algorithms.
- Created command-line tool for applying three deblurring algorithms, utilizing sparse matrices for efficiency.
- Solely executed the majority of the project's implementation work, exceeding expectations.
- Applied 3D rendering expertise for accurate satellite to earth frame coordinate conversion.

`jupyter notebooks` · `python` · `numba` · `pyvista` · `numpy` · `xarray` · `matplotlib` · `scipy` · `sparse matrices`

Nov. 2022 - Dec. 2022
2 months
Client: EUMETSAT

MTG-LI Off-Line Tools - Stage 2 (RESPASE Task 8)

Service Level Manager / Technical Lead Engineer

- Adapted and evolved L1 and L2 satpy readers to align with EUMETSAT's evolving product format.
- Refactored code, separating level 1 and level 2 interfaces to enhance end user usability.
- Unified data pipeline daskification for optimization and readers consistency
- Introduced user-configurable filters in LI-PVT events ingestor.
- Fixed GUI issues, including animation function and UTC time synchronization.
- Delivered updated satpy readers and LI-PVT GUI application following best practices.
- Successfully managed two GitLab repositories for concurrent development needs.

`python` · `pyQt5` · `vispy` · `xarray` · `satpy` · `dask` · `gitlab`

Jun. 2022 - Dec. 2022
7 months
Client: EUMETSAT

FCI-ATT v1.0 implementation (RESPASE Task 7)

Service Level Manager / Technical Lead Engineer

- Implemented MTG FCI-ATT application in Python, supporting CLM and FIR algorithms.
- Developed GUI for parameter tuning, spectral test selection, and complex algorithm computation.
- Optimized performance with GPU computing and numba's JIT C++ acceleration.
- Supported features like region of interest, cross-section display, histogram computation, layer transparency, etc.
- Conducted comprehensive unit testing using pytest.
- Maintained code consistency with blake, flake8, isort, pylint.
- Build tool documentation in ReStructuredText (rst) with sphinx.

`python` · `pyQt5` · `glsl` · `vispy` · `xarray` · `satpy` · `numpy` · `yaml` · `dask` · `numba` · `esbonio` · `sphinx`

Nov. 2021 - Mar. 2022
5 months
Client: EUMETSAT

MTG-LI Off-Line Tools - Stage 1 (RESPASE Task 6)

Service Level Manager / Technical Lead Engineer

- Responsible for features implementation, GitLab repository maintenance, and communication with the client.
- Developed satpy readers for SEVERI and FCI instrument inputs, enabling lightning event processing.
- Implemented ingestor utilities for background imagery and lightning events pre-processing.
- Extended the Python tool LI-PVT using xarray to import netCDF files and display their content.
- Enhanced the LI-PVT GUI application with filtering options and background selection support.
- Utilized Cartopy for coastline generation and optimized performance using multi-threading techniques.
- Conducted unit tests, including dummy input generation with pytest.

`python` · `pyQt5` · `vispy` · `xarray` · `satpy` · `netcdf` · `cartopy` · `yaml` · `pandas` · `pillow` · `matplotlib` · `pytest`

Dec. 2015 - Present
7 years, 10 months
Client: L3 Wescam

SimCore

Lead Architect & Developer

- Led the development and architecture of SimCore, a cutting-edge simulation system, as the successor to the MX-Sim project for L3 Wescam.
- Successfully delivered the project within a challenging timeframe, utilizing state-of-the-art technologies.
- Adapted the system to accommodate changes in the underlying 3D platform, including transitioning to 64-bit architecture and addressing compatibility issues during the upgrade from Virtual Battlespace 2 (VBS2) to VBS3.
- Introduced a higher level of abstraction, making the core system "3D agnostic" and encapsulating DirectX interactions through a plugin.
- Significantly enhanced long-term maintainability through the development of a unified architecture, merging five separate software components into a single cohesive system.
- Achieved major optimizations by migrating from Lua and Java code to C++ and JavaScript, resulting in improved performances.
- Implemented an extensive test-driven development approach, writing over 1500 complex tests to ensure robustness and reliability.
- Followed strict version control practices using git/gitlab workflows for seamless collaboration.
- Developed multiple communication channels, including sockets, serial ports, and IPC, to facilitate efficient data exchange.
- Built support for SDI video streaming with embedded KLV data, enhancing the capabilities of the simulation system.
- Designed and integrated user interfaces and heads-up displays (HUDs) as web applications using AngularJS framework, seamlessly integrated with the engine using Chromium Embedded Framework 3 (CEF3).
- Achieved remarkable results within a year, refactoring and optimizing the work that had previously taken a team of seven qualified developers more than four years to accomplish.
- SimCore now in production phase, providing full support and improved training environment for all Wescam turret models.

`C++` · `VBS3` · `VR-Vantage` · `DirectX 9` · `DirectX 11` · `HLSL` · `NodeJS` · `CEF3` · `AngularJS` · `HTML5`

Nov. 2020 - Sep. 2021
11 months
Client: EUMETSAT

GUI_CM

Senior Software Engineer

- Played a key role in enhancing the graphical user interface (GUI) of the GUI_CM tool, an integral component of the Cal/Val toolset for Level 2 (L2) products from the EPS-SG Visible/Infrared Imager (VII) at EUMETSAT.
- Collaborated closely with a team of two developers to upgrade and improve the GUI_CM software written in IDL (Interactive Data Language).
- Unified the input files used by the software, ensuring consistency and streamlined data processing.
- Implemented the capability to read multiple new variables from NetCDF files.
- Developed functionalities for building new RGB images, providing users with improved visual representations of data.
- Extended support for additional test types, specifically restoral tests, enabling more comprehensive analysis of METimage L2 cloud mask.
- Implemented support for polygonal Region Of Interest (ROI) construction and usage, allowing focus on specific areas on Earth.
- Integrated scene-specific test threshold tuning, providing flexibility in adjusting parameters for different scenarios.
- Designed and implemented functionality to build, save, and load Constructed Reference Mask (CRM) for advanced comparisions.
- Made significant updates and improvements to the GUI itself, enhancing user interactions and usability.
- Utilized IDL, Python, and NetCDF extensively throughout the development process.
- Utilized Linux and SSH for remote access to EUMETSAT workstations.

IDL · GUI design · linux · ssh · gitlab · netcdf

Nov. 2015 - Apr. 2016
6 months
Client: EUMETSAT

GRAS PPF

Senior Software Engineer

- Contributed as a key member in developing the GRAS PPF software, processing raw data from METOP satellites for atmospheric sounding.
- Ported and integrated the Full Spectrum Inversion (FSI) algorithm from Python to C++, enhancing the "wave optics" processing.
- Implemented core methods, leveraging Python models to optimize software functionality and performance.
- Developed comprehensive unit tests using Boost Test framework to ensure accurate and isolated behavior of implemented methods.
- Created Python bindings and utilized nosetest for robust comparison between C++ and Python code bases.
- Enhanced Python test procedure, enabling granular validation of C++ code by breaking complex computations into smaller steps.
- Streamlined integration of C++ and Python code using Boost Python framework, improving efficiency and reliability.
- Practiced Scrum methodology, contributing to regular sprints and project milestones.
- Implemented testing improvements, reducing debugging cycles and increasing efficiency of acceptance tests.

C++ · AIX · xlC · Python · nosetest · Boost Test Framework · Gitlab · Boost Python · Scrum

Apr. 2015 - Dec. 2015
9 months
Client: EUMETSAT

ASCAT PPF

Senior Software Engineer

- Successfully executed the porting of the software code base from AIX platform to Linux.
- Ensured cross-platform compatibility and enhanced performances.
- Implemented improvements to the data processing pipeline, dynamically generating necessary auxiliary data on-the-fly, eliminating the need for undesired precomputation stages.
- Optimized the multithreading structure of the software, leveraging a custom thread pool mechanism for performance enhancements.
- Replaced AIX/xlC-specific library with an open-source multi-platform alternative (FFTW).
- Ensured platform compatibility by considering endianess and updating the code base to support both gcc and xlC compilers.
- Conducted rigorous regression tests against the reference implementation, validating algorithm changes and ensuring accuracy.
- Developed a high-speed tool in LuaJIT from scratch to process ASCAT PPF outputs, surpassing the EUMETSAT tool by a factor of x100.
- Generated preprocessing "NTB data" in real-time, eliminating the need for a dedicated preprocessing stage.
- Achieved remarkable multithreaded performance improvements by unifying the multithread handling systems in the software and replacing OpenMP with a custom thread pool mechanism.
- Exceeded expectations by delivering successful completion of all tasks and achieving significant multithreaded performance improvements, enabling scalable speed gains on target stations.

C++ · AIX · xlC · Boost · Threading · Profiling tools · Lua · Latex · shell

INSYEN AG · Software Engineer · Jan. 2009 - Oct. 2015

Jun. 2015 - Feb. 2016
9 months
Client: EMT

UVIS
Software Architect & Developer

- Developed the uVIS project, a simulation system with 3D visualization capabilities that integrated with the EMT's in-house Unmanned Aerial Vehicle (UAV) simulator, built on top of Bohemia Interactive Simulations Virtual Battlespace 3 (VBS3).
- Redesigned all Lua-based classes, applying appropriate object-oriented programming (OOP) patterns, resulting in improved code structure and maintainability.
- Developed more than 180 behavior tests to ensure the application's functionality and behavior, enabling early validation and identification of issues during the development phase.
- Replaced the built-in support for the DIS protocol in VBS2 with a custom re-implementation in Lua, addressing limitations and bugs encountered in the initial implementation.
- Designed and created a dedicated user interface using IUP, facilitating efficient testing and validation of all supported system commands, including UAV creation, entity placement, weather control, HUD display, and waypoint management.
- Implemented a robust mechanism to initiate and control VBS3 scenarios across multiple stations from a single master instance, enhancing the system's efficiency in handling network scenarios.
- Despite joining the uVIS project when it was already behind schedule, the introduction of behavior tests and subsequent efforts led to the development of a stable and bug-free system, which was successfully delivered and validated by EMT.

VBS2 · C++ · Lua · DirectX 9 · HLSL · DIS · CMake · IUP

Jan. 2012 - Apr. 2016
4 years, 4 months
Client: L3 Wescam

MX-SIM
Main 3D Developer

- Developed the 3d visualization component for the MX-Sim simulator project, a comprehensive software suite designed to simulate various Wescam MX-Series turrets.
- Collaborated with a team of 7 professionals to develop this system, which included a core simulator implemented in Java and a dedicated plugin, named VBSSim, for the Bohemia Interactive Simulations Virtual Battlespace 2 (VBS2) engine.
- Developed additional components to facilitate connections with official turret controllers, lasers arming/firing protocols, and SDI video streaming with embedded KLV data.
- Designed and implemented the VBSSim plugin, enabling seamless integration between the MX-Sim system and the VBS2 engine.
- Created a low-level DirectX hook system to extend the capabilities provided by the VBS2 engine, enhancing the overall functionality and realism of the simulation.
- Developed a custom Lua binding generator tool (sgtLuna), utilizing doxygen XML parsing to generate advanced Lua bindings in C++ for complex APIs such as DirectX, VBSFusion, and OpenSceneGraph.
- Utilized object-oriented programming (OOP) principles in Lua to construct models for essential turret components, including gimbals, sensors, cameras, filters, illuminators, designators, overlay sub-systems, and menu sub-systems, ensuring accurate behavior.
- Implemented complex heads-up displays (HUDs) with DirectX and advanced post-processing effects using HLSL, including depth of field, noise grain, thermal simulation, camera calibration processes, exposure adaptation, image blending, image sharpening, and more.
- Developed the network communication layer, incorporating socket communication with Wescam-specific protocols, within the VBS2 plugin to enable seamless interaction between the MX-Sim core simulator and the plugin.
- Through successful development efforts, the MX-Sim product has been adopted by numerous Wescam customers for effective training on the MX-Series turrets.

VBS2 · C++ · Lua · DirectX 9 · HLSL · PhotoShop · CMake · Latex

Oct. 2012 - Dec. 2012
3 months
Client: L3 Wescam

E-LEARNING
Main Developer

- Developed and added maritime scenarios to Wescam e-Learning software, enhancing user training.
- Restructured ActionScript core simulation classes with improved Object-Oriented Programming.
- Optimized assets and scenes for smooth performance on low-end devices.
- Delivered project milestones on time, validated by Wescam.
- Fine-tuned software performance and optimized visual assets for optimal learning.

ActionScript · PhotoShop · FlashDevelop IDE · Away3d · 3DS Max

Nov. 2009 - Jan. 2011
1 year, 3 months
Client: ESA

U-CORVS
Main Developer

- Debugged and resolved software issues, ensuring smooth operation of the U-COVRS video streaming application.
- Streamlined and simplified codebase through strategic refactoring to maintainability.
- Enhanced web interface layout based on user requirements, optimizing user experience.
- Met tight project schedule by promptly addressing and resolving submitted issues.

C++ · PHP · gSOAP · Java · SLES · VLC · shell · cmake · CSS · SVN

Jan. 2009 - Dec. 2011
3 years
Client: DLR

NG3DV

Software Architect & Developer

- Developed the NG3DV application, a project aimed at re-implementing and enhancing the SatViz simulation system.
- Defined and initiated the complete software architecture for NG3DV, ensuring a unified and simplified approach.
- Upgraded the 3D engine from SatViz and seamlessly integrated it into a wxWidget-based application.
- Acquired expertise in GUI development, networking protocols, project management, OpenSceneGraph framework, GLSL shaders, and online Earth Map services.
- Implemented advanced features on the 3D engine, including retrieval of Microsoft Virtual Earth imagery, complex shading effects, orbit propagation, and video streaming with ffmpeg.
- Integrated NG3DV in web pages using a Java applet and implemented license management and encryption.
- Successfully replaced SatViz installations with NG3DV software, exceeding initial expectations and enabling precise positioning and monitoring of over 1000 satellites concurrently.

C++ · OSG · OpenGL · GLSL · Java · 3DS Max · Visual Studio · CMake · SVN

SESS GbR · Junior Software Engineer · May 2005 - Dec. 2008

May 2005 - Dec. 2008
3 years, 8 months
Client: DLR

SATVIZ

Main 3D Developer & Modeler

- Contributed as a 3D Developer and Modeler in the SatViz project, a real-time satellite simulation tool with advanced 3D visualization.
- Designed and implemented a powerful 3D engine using OpenSceneGraph, specializing in spatial mission modeling.
- Developed expertise in C++ programming, 3D space algebra, OpenGL pipeline, OpenSceneGraph framework, and 3D modeling techniques.
- Actively engaged with the OpenSceneGraph community, creating plugins for various file formats and actively contributing to the project.
- Acquired multi-platform experience by working with Visual Studio on Windows and Eclipse on Linux.
- Optimized 3D models using 3DS Max to ensure compatibility and performance within the 3D engine.
- Utilized UML diagrams to design classes and systems, facilitating effective software development.
- Created Windows installer construction scripts using NSIS for seamless deployment of the SatViz software.
- The SatViz software achieved successful use in public demonstrations at DLR, showcasing its capabilities for several years.

C++ · OSG · Java · JNI · 3DS Max · Visual Studio · Eclipse · SCons · NSIS · SVN

Personal Projects

Apr. 2022 - Present
1 year, 6 months

NervLand

- Current main personal project aiming at conducting experiments on advanced 3D rendering techniques.
- Also storing multiple sub-projects and experiments, both 3D rendering and finance/game/utility-oriented.
- Intensively working on WebGPU tests, implementing a C++ WebGPU engine (for Windows and linux).
- Creating an automated mechanism for generating Lua bindings without relying on Sol library, using LLVM for parsing C++ sources.
- Building working bindings for various large libraries, including Clang, Qt6, and Vulkan.
- Further exploring just-in-time compilation with LLVM
- Investigating web assembly support to develop a 3D game engine that can run in both web browsers and desktop environments.
- Leveraging Clangd extension for Visual Studio Code to format, clean and maintain C++ code efficiently.
- Building applications within NervLand, including CryptoView, HomeView, and Vulkan application showcasing 3D rendering capabilities.
- Written extensive code base for NervLuna, to robustly generate Lua bindings, even including support for C++ templates.

C++ · Python · Lua · WebGPU · Google Dawn · Web Assembly · LLVM · MSVC · Vulkan · OpenGL · Boost Test · Emscripten · clangd · Clang-Tidy · Clang-Format · SDL · Visual Studio Code · Qt6 · Curl · Lua binding generation · AST

Apr. 2022 - Present
1 year, 6 months

NervHome

- Developed and managed various configurations, scripts, and components for different projects, including ASCAT deblurring task, EUMETSAT RESPASE Tasks and SimCore project.
- Implemented configuration files for Python setup, scripts for testing and running work projects in specific environments.
- Developed scripts for media file management, including renaming, deduplication, classification, and format conversions.
- Created applications for media display, XT4 camera raw image processing,
- Created python domotic applications for swimming pool and solar heating system control.
- Worked on numerous blockchain-related projects, including advanced Solidity contracts and interaction with multiple protocols.
- Built a working optimized Flash Loan arbitrage system for EVM compatible blockchains using solidty assembly code for reduced gas cost.
- Conducted dynamic operations for monitoring gas prices, collecting blockchain data, building large datasets statistics.
- Tested smart contracts using the Ganache environment.
- Developed components for backup management, disk monitoring, PostgreSQL database backup, and file operations.
- Implemented various Python classes, such as password generator, database management, text translator, etc.
- Developed reusable GUI elements for the Qt framework as well as a Task management mechanism
- Implemented GPIO interfacing for Raspberry Pi devices.
- Created numerous simple tools for GIF/image processing, video conversion, media scraping, and Twitter communication.

Python · ffmpeg · PostgreSQL · SQLite · SQL · PyQt5 · Ganache · GPIO · OpenGL · Matplotlib · Twitter API · Let's Encrypt · Shell scripts · Jupyter · ModernGL · Solidity · EVM blockchain · Uniswap protocol · Solana blockchain · Numpy · psycopg2 · Web3 · Pillow · Ganache · DeFi protocols · Flash Loan arbitrage · Backup management system · libraw · Jupyter books · GLSL

Mar. 2022 - Present
1 year, 7 months

NervProj

- Developed NervProj as a standalone dependency management system for NervLand project.
- Implemented Python-focused components and a Python context object for task execution.
- Created modular components for administration, AI, compilations, environments management, etc.
- Built various dependency builders for libraries like Boost, LLVM, Qt6, and more.
- Supported multiple compilers and platforms, including Windows, Linux, and Emscripten.
- Integrated tools for networking, image manipulation, and GUI development.
- Designed subproject functionality for incorporating external scripts and components.
- Managed project build processes, including CMake, EM SDK, and Git operations.
- Developed support for Raspberry systems for home automation applications.
- Implemented C++ test project with Python-C++ binding generation using Boost Python.
- Configured tools, libraries, scripts, and Python environments through a comprehensive configuration file.
- Utilized various packages for media handling, image processing, and AI tasks.
- Created scripts for CV generation, audio-to-text conversion, media manipulation, and more.
- Ensured compatibility with CUDA acceleration for AI-related tasks.
- Integrated a wide range of tools and technologies to enhance project development and automation.

C++ · Python · OpenAI Whisper · YAML · stable diffusion · boost · clang · MSVC · Emscripten · web assembly · FFmpeg · CMake · Git · Boost Python · NodeJS · request · moviepy · OpenCV Python · torch · CUDA · pillow · odfpy · fontawesome

Sep. 2017 - Apr. 2022
4 years, 8 months

NervSeed

- Developed a comprehensive collection of bash and shell scripts for various administrative tasks
- Implemented Docker-related scripts and explored Docker containerization
- Developed media tools for processing family videos and pictures
- Created project-specific scripts for efficient project navigation and version control
- Gained experience in Arduino programming for domotic elements
- Focused on C++ programming and built a core project for 3D engine components
- Explored DirectX 12, OpenGL, Vulkan, and Unity for rendering and engine development
- Generated Lua bindings for various C++ libraries and experimented with just-in-time compilation using LLVM
- Built Python modules and applications for cryptocurrency, finance, and DeFi protocols
- Developed a GUI application for cryptocurrency price monitoring and analysis
- Conducted experiments on neural network style transfers and arbitrage in the cryptocurrency market
- Worked with Solidity and Ethereum smart contracts for advanced financial operations
- Explored Django framework for web application development and backup mechanisms
- Conducted deep learning experiments using TensorFlow and Keras for image recognition
- Automated participation in decentralized finance protocols on various EVM-compatible blockchains
- Interfaced with different protocols through direct requests and web3 Python module
- Utilized wxWidget and Qt for GUI applications development.

shell scripts · docker · linux · ffmpeg · arduino · c++ · cmake · Lua · 3d math · DirectX 12 · Vulkan · OpenGL · LLVM · OCCT · Sol3 · python · PostgreSQL · pyQt5 · matplotlib · requests · Solidity · smart contracts · Django · TensorFlow · Keras · Deep learning · Binance Smart Chain · Ethereum · Polygon · Solana · REST communication · web3.py · RPC · wxWidget · Qt6

Education and Training

Sep. 2002 - Dec. 2005
3 years, 4 months

Ecole Centrale Paris

Master's Degree

- Master's Degree with major in Applied Physics
- Successfully completed the final examinations with distinction.

Sep. 2000 - Aug. 2002
2 years

Louis-Le-Grand, Paris

Preparatory Classes

- Rigorous training in advanced mathematical concepts and physics principles

Main Skills

General Programming

- C & C++
- Python
- Lua
- Solidity
- Rust

Data Science

- Numpy & pandas
- satpy
- dask
- scipy
- R

Machine Learning

- torch
- tensorflow
- ggml

GPGPU & 3D Programming

- CUDA
- OpenCL
- DirectX 9 & 11
- DirectX 12
- Vulkan
- WebGPU
- HLSL
- GLSL
- WGSL

Web Development

- Javascript & Typescript
- HTML5
- CSS & SaSS
- NodeJS
- MongoDB
- Express
- AngularJS

3D Engines

- Virtual Battlespace 3
- Virtual battlespace 4
- VR-Vantage
- Unity 3D
- Unreal Engine 4
- Lumberyard

Finance & Trading

- MetaTrader
- MQL4
- MQL5

Virtualization

- Docker
- VMWare Workstation
- ESXi

Others

- Photoshop
- 3DS Max
- Latex

Other Skills

- **System Administration:** Proficient in maintaining multiple local servers, including webserver (nginx), GitLab server, Rocket.Chat server, Jupyter server, Kimai2 server, mail server, WordPress server, Nextcloud server, and more.
- **DIY Enthusiast:** Skilled in woodworking and metalworking, with a keen interest in electronic projects and building domotic installations.
- **Violin Learning:** Currently pursuing the challenge of learning to play the violin.

Interests

- **Cooking:** Enjoy exploring culinary arts and experimenting with new recipes.
- **Hiking in the Mountains:** Find solace and inspiration in nature, particularly through hiking adventures.
- **Fitness Enthusiast:** Regularly engage in running, cycling, and bodybuilding to maintain an active and healthy lifestyle.
- **Gaming:** Passionate about video games and fascinated by the immersive experiences they offer