Ethan Sheehan

+44 7920 575 483 | ethansheehan9@gmail.com | fc23871@bristol.ac.uk | linkedin.com/in/ethan-sheehan

Education

MEng Aerospace Engineering, University of Bristol — Bristol, UK
First-Class Honours (predicted) — *2023 – 2027*

- **Roderick Collar Prize 2024/25**, awarded for combining academic excellence with extracurricular contribution.
- Third-Year Research Project: "Control System Design for a Landing Burn of a Thrust-Vector-Controlled Rocket after Main Chute Deployment."

A-Levels, British International School of New York — New York, USA
*A*A*A: Mathematics, Further Mathematics, Physics, Chemistry* — *2021 – 2023*

Experience

Computer Vision Research Intern — Jul. 2026 – Aug. 2026
Rolls-Royce UTP, Institute of Lightweight Engineering (ILK), TU Dresden — *Dresden, Germany*

- Built a real-time AI crack-tip detection system for composite fracture testing, running entirely on an **NVIDIA Jetson AGX Orin**.
- Selected for the **Rolls-Royce** Composites University Technology Partnership student exchange between the Bristol and Dresden UTCs, embedded in the ILK's composite-testing research group.
- Designed a two-stage cascade of **PyTorch** CNNs (coarse tip localisation at 512 px, then fine localisation at native resolution inside an uncertainty-sized crop) to deliver both high capture rate and high precision on 5 MP frames.
- Achieved 0.35 mm median crack-tip error, with the cascade beating the single-stage baseline on every cross-validation fold (1.4× to 16.5× accuracy gain) and the crop hand-off within ±1.1 px of an oracle.
- Deployed models via **ONNX** and **TensorRT** FP16 on the Orin's tensor cores, ∼9× faster than eager PyTorch at matched accuracy (∼24× on a ResNet20 benchmark), sustaining 20-25 fps live across two 5 MP cameras.
- Modelled aleatoric and epistemic uncertainty so every detection ships with a confidence (σ) and a no-tip probability, keeping unreliable frames out of downstream fracture-mechanics calculations.
- Integrated the full test-rig stack: dual Allied Vision USB cameras free-running, with load and displacement channels read separately through a 16-bit ADC over **I2C** at 200 Hz per channel (60 s proof run: 11,998 samples, zero dropped).
- Developed a **React** + **FastAPI** dashboard streaming both cameras with live per-frame tip, uncertainty and crop overlays over **WebSockets**, exposed for remote monitoring through a **Cloudflare Tunnel**.
- Curated and labelled a 5,495-frame dataset across 11 test sequences and 6 carbon- and glass-fibre specimens; designed leave-one-specimen-out cross-validation with frame guard bands to prevent near-duplicate leakage.
- Established each fold's labelling floor with an oracle-crop study (down to ∼3 px on clean folds, ∼7 px ≈ 0.35 mm on the DIC-speckled fold), separating data limits from model limits to direct future work.
- Delivered the documented multi-module codebase (capture, DAQ, vision, app) on the institute's GitLab and presented the system to the research group in an end-of-internship technical talk.

Propulsion Lead — Feb. 2025 – Present
HyPower Bristol (University of Bristol's premier rocketry team) — *Bristol, UK*

- Lead the 80+ student propulsion effort for Bristol's entries in the European Rocketry Challenge (Europe's premier rocketry competition: 30 universities, 16 countries) and UK Race to Space (the world's largest student propulsion competition).
- Hot-fired the world's first **CP1-Aluminium** rocket engine (an additively manufactured high-performance aluminium alloy).
- Designed the world's first **CP1-Aluminium** rocket engine in partnership with the Manufacturing Technology Centre, applying **DfAM** principles and developing depowdering solutions so its complex internal cooling channels printed powder-free; hot-fired to 2.1 kN at UK Race to Space 2025.
- Secured £120,000+ in funding and sponsorship, co-leading the pitches and sponsor relationships that financed both national and European campaigns.
- Designed an Electronic Pressure Regulator with an adaptive dual-cascade **PID** controller, achieving real-time active tank-pressure regulation of a 300 bar feed system and independent throttling of each propellant via SRAD servo-actuated valves.

- Led the second-ever integrated hot-fire campaign in UK university rocketry to flight-qualify the EuRoC vehicle: 3 successful burns, 5.5 kN peak thrust, 40+ kNs total impulse, zero anomalies.
- Built a **Simulink/Simscape** digital twin of the full 300 bar feed system, modelling two-phase nitrous-oxide/nitrogen multi-fluid dynamics with a **Stateflow** sequencer for fuelling, venting and firing procedures.
- Validated the fluids model against hot-fire test data and used it for software-in-the-loop testing of the pressure-regulator controller, to write safety procedures, and to troubleshoot live on test days.
- Oversaw development of the 300 bar feed system: Swagelok plumbing, SRAD servo-actuated ball valves, an SRAD pressure manifold, a coaxial propellant tank, and passive relief sized to competition safety requirements.
- Designed and tested a novel Hydrogen/Oxygen torch ignition system, fired 10 times across a range of inlet pressures to prove the concept.
- Developed a canned-motor electric fuel pump with a Barske-style impeller using **AxSTREAM** and **CFturbo**.
- Architected **MATLAB** pipelines to post-process .h5 hot-fire telemetry; the analysis set the requirements for each engine design iteration.
- Wrote the risk assessments, test plans and safety cases under which a student team is permitted to run 300 bar hot-fire operations; maintained assembly procedures and flight-hardware inventory.
- Ran a validated **OpenFOAM** CFD campaign on the ARIA supersonic nose cone (Mach 1.8), predicting drag, surface pressures and thermal loading for Bristol's first supersonic rocket (full write-up under Projects).
- Generated **Fusion 360** CAM toolpaths and machined a nose-cone layup mandrel on a CNC router.
- Designed an isogrid combustion-chamber structure to minimise engine mass for the team's first LOx engine.
- Bristol finished top UK team at EuRoC 2025, ahead of Imperial College London and the University of Bath, scoring highly on vehicle design and the technical report.
- Contributing to Launch 4 Change, an ARIA-funded programme (10 UK universities selected) mentoring 5 schools for the UK Youth Rocketry Challenge while developing Bristol's first supersonic rocket.

Composites & Machine Learning Research Intern Jun. 2025 – Aug. 2025

Bristol Composite Institute (world-leading composites research centre) *Bristol, UK*

- Ran a hands-on experimental campaign on 3D-printed hygromorph bio-composites (materials that self-shape with moisture), owning the full loop from specimen fabrication to a trained prediction model.
- 3D-printed, moisture-conditioned and tensile-tested 80+ specimens to failure on a **Shimadzu** testing machine, recording each test on high-resolution video.
- Imaged specimen microstructure with **Zeiss** optical microscopy and quantified hygromorph curvature, building curvature-stress datasets and shape-analysis overlays.
- Post-processed tensile-test video and microscopy imagery into a labelled training dataset.
- Built and trained a convolutional neural network (**TensorFlow**, Python) on the collected data to predict the material properties of the printed composites straight from images, replacing exhaustive physical testing.
- Delivered the trained model plus the reusable labelled dataset to the research group.
- Continuing the research independently, rebuilding the pipeline in **PyTorch** to predict curvature and surface-stress plots from microscope images of print defects.

Avionics Demonstrator May 2025 – Jun. 2025

Bristol Flight Lab (University of Bristol's centre for multidisciplinary aerial robotics) *Bristol, UK*

- Built the avionics teaching resources for Bristol's AVDASI 2 design course, in which 60-student teams design, build and fly 1.2 m-wingspan UAVs.
- Wrote flight-controller software in Python using **MAVLink** and I2C to log and plot live telemetry from custom sensors.
- Developed an interactive **React** GUI to remotely actuate the UAV's control surfaces over WiFi and radio.
- Wrote modular Python/LUA avionics example code for Ardupilot and the MATLAB UAV Toolbox, and published full documentation with MkDocs and GitHub Pages.
- Assembled and tested ESD-sensitive flight electronics, and built a demonstrator UAV that lecturers use to guide student designs.

Engine Lead Nov. 2023 – Jan. 2025

HyPower Bristol (University of Bristol's premier rocketry team) *Bristol, UK*

- Designed and test-fired a 5 kN regeneratively cooled liquid bipropellant engine through 3 full-duration burns, driving it to 6 kN (20% past design conditions) with zero hardware damage: the **UK student thrust record** at the time.

- Led design and construction of the additively manufactured **Inconel 718** combustion chamber and nozzle (Autodesk Inventor), iterating the CAD to meet DMLS print constraints and smooth internal flow through the manifold and cooling channels.
- Wrote Python/MATLAB scripts to size the hoop and thermal stresses, engine pressure drops, and the coaxial propellant tank.
- Verified the cooling channels and manifold in **Abaqus CAE** FEA under combined pressure, thermal and structural loads, reaching a minimum safety factor of 3.24.
- Conducted CFD analysis of the regenerative cooling system to characterise fuel-manifold efficiency.
- Helped manufacture the team's house-designed servo valves and assemble the 300 bar feed system.

Selected Projects

Supersonic Nosecone CFD — V&V Campaign for the ARIA Rocket | *OpenFOAM, Python*

- Ran a validation-first CFD campaign (**OpenFOAM**, density-based solvers) for supersonic flow over the ARIA rocket nose cone at Mach 1.8, predicting surface pressures, thermal loading, shock structure and drag.
- Validated the methodology against two NACA wind-tunnel datasets and exact compressible-flow theory (Taylor–Maccoll, shock-expansion): mean $|\Delta C_p| = 0.0040$, within the tunnel's quoted ± 0.004 uncertainty.
- Quantified grid-convergence (GCI), farfield and numerical-scheme sensitivity; bounded nose wave drag at $C_d = 0.0364 \pm 8\%$ and total drag $C_d = 0.197$ at flight Reynolds number, with a 12.7× speed-up from local time stepping.

Magic Carpet — Rocket-Engine Design Platform | *Python, Cantera, FastAPI, React, Docker*

- Built a full engine design and optimisation platform: thermochemistry (**Cantera**), regenerative-cooling and combustion-stability analysis, structural life, and automated CAD/report generation.
- Engineered as production software: Python core behind a **FastAPI** REST API with a **React** frontend, containerised with **Docker** and deployed to the cloud.
- Held to ∼2,400 automated tests and 84% coverage on CI.

Grid-Sentinel — Multi-Agent Drone-Swarm C2 | *Python, FastAPI, OpenCV, React, Cesium*

- Built a decision-centric command-and-control platform for drone swarms with multi-sensor fusion and human-in-the-loop approval workflows.
- Developed a 3D-globe operational frontend (React + **Cesium**) over a FastAPI backend with OpenCV-based perception.
- Maintained ∼1,900 automated tests on a CI matrix (GitHub Actions).

Line-Following Drone — RL path planner on a Parrot Mambo | *MATLAB, Simulink* Jul 2026 – Aug 2026

- Behaviour-cloned a policy from 21k expert samples over 300 domain-randomised surrogate tracks, refined with soft actor-critic, then deployed unchanged to the real Simulink model, completing **all 12 unseen tracks** at 45 mm cross-track RMS and 14 mm mean landing miss.
- Replaced the hand-tuned path planner with an 11-input, two-output policy whose tanh output layer bounds each position increment to 0.36 m, so the fixed PID loop can never be commanded into the arena wall.
- Beat the team's conventional controller 12 of 12 to 0 of 12 on identical blind tracks, with cross-track RMS 45 mm against 158 mm (their two failures traced to a dead heading-memory line and a NaN reaching the airframe).
- Shipped the policy as baked weights evaluated by three matrix multiplies inside Simulink, code-generating to C for the Parrot target, behind 77 automated checks covering dynamics, vision front-end and baking equivalence.

Pokémon TCG AI Battle Challenge — Kaggle × The Pokémon Company | *PyTorch* Jun 2026 – Aug 2026

- Placed **351st of 6,807 teams** (top 5%) as a two-person team with a behaviour-cloned ensemble of three ∼500k-parameter PyTorch networks trained on top-ladder replays, under a two-turn minimax search override.
- Wrote a replay scraper (training data and exact opponent decklists), behaviour-cloned opponents per archetype, and an SPRT and Wilson-interval harness that overturned six decisions, including a 140k-game RL line.
- Built the F6 decision harness: anchored gauntlet with Wilson 95% intervals and SPRT early stopping, side-balanced cells, common-baseline scoring that cut the noise floor from ±2.1 pp to ±1.25 pp with no new games.
- Scraped and parsed top-ladder episodes into F4-schema parquet, recovering opponents' exact 60-card lists from replay card serials (60/60 on every episode) and a band-weighted meta deck-frequency table.
- Implemented K1 policy-guided PUCT search (65.5% vs baseline) and K2a own-turn-horizon search with a frozen heuristic leaf and 2-ply retaliation term, with leak tests.
- Ran under Kaggle's 1.6 vCPU / 600 s-per-game budget with a never-crash shell: zero crashes across every gate reported.

IMC Prosperity — Algorithmic Trading Competition | *Python, Monte-Carlo backtesting*

- Built a custom data-generating-process simulator plus a backtesting and replay toolchain calibrated to the competition's market microstructure.
- Developed market-making and mean-reversion strategies, optimising parameters against simulated PnL and risk metrics across rounds.

Additional

BAE Systems Capture-the-Flag Competition Feb. 2024

- Solved cybersecurity challenges in a team: RSA decryption, web exploitation, SQL injection.

Technical Skills

Programming: Python, C++, MATLAB, Simulink, C#/.NET, JavaScript/TypeScript, SQL, LUA, Bash
ML & Data: PyTorch, TensorFlow, OpenCV, NumPy, pandas, SciPy, CNNs, reinforcement learning (SAC/TD3/PPO), Monte-Carlo simulation
Software & Infra: FastAPI, React, Node.js, Docker, Git, Linux, CI/CD (GitHub Actions), pytest, REST, WebSockets, Cloudflare Tunnel, cloud deployment
Engineering & CAE: Autodesk Inventor, Fusion 360 (+CAM), CATIA, Siemens NX, Abaqus CAE, Ansys, OpenFOAM, Simscape/Stateflow, Cantera, RPA, AxSTREAM, CFturbo, GMAT, NASA CEA
Embedded & Avionics: NVIDIA Jetson, Ardupilot, Arduino, Raspberry Pi, MAVLink, I2C
Tools: Microsoft Office Suite, Google Workspace, GitHub Projects, LaTeX
Languages: English & French (fluent), Spanish (proficient); Canadian citizen