

Anant Saraf

Providence, 02912 | Phone: (401) 259-4181 | E-Mail: anant_saraf@brown.edu | [LinkedIn](#) | [Photography](#) | [GitHub](#)

EDUCATION

Brown | RISD Dual Degree Program GPA 3.98/4.0

Providence, RI | Expected Graduation May 2027

Brown University, B.A. Computer Science

Relevant Coursework: Computer Vision, Deep Learning, Intro to Computer Graphics, Computational Photography, Intro to Software Engineering, Linear Algebra, Discrete Structures and Probability

Rhode Island School of Design (RISD), B.F.A. Photography

EXPERIENCE

Undergraduate Teaching and Research Award, Geospatial Object Detection Pipeline **Providence, RI | June – Aug 2025**

- Designed and implemented an end-to-end SAR computer vision pipeline in Python to detect ocean eddies in open water and marginal ice zones; packaged as a reproducible, config-driven workflow usable by non-technical domain experts.
- Built GeoTIFF ingestion via API, SAR-specific preprocessing, YOLO inference, and geospatial post-processing to produce analysis-ready outputs for downstream analysis.
- Fine-tuned YOLOv11 with SAHI sliding-window inference on a 3,000-image dataset (including null samples), achieving approximately 90% precision, 78% recall, and 83% mAP@0.5 on a held-out test set.
- Partnered with 2 domain experts to translate scientific objectives into dataset and modeling decisions; monitored training and created dataset curation tools to support iteration and handoff.

Joint Undergraduate-SRC Teaching Assistant, Intro to Software Engineering (CSCI 0320)

Providence, RI | Fall 2025

- Design course material covering API development, socially responsible computing (SRC), software architecture, data validation, version control workflows, and agentic AI-assisted programming.
- Led weekly labs and office hours for 100+ students, mentoring teams, evaluating project progress in sprint demos, and providing actionable feedback to enhance learning outcomes.

JISIASR, Medical Imaging Research Intern (Deep Learning)

Kolkata, India | July 2024 – Sept 2024

- Developed a repeatable training and evaluation pipeline for lung segmentation on chest X-rays using U-Net variants, working under small-data and label-quality constraints.
- Ran controlled experiments on preprocessing and training choices (e.g. CLAHE, augmentations, early stopping) to reduce overfitting and improve validation behavior; compared U-Net variants and summarized results for handoff.
- Implemented a ResNet-50 classifier baseline and applied Integrated Gradients to probe model sensitivity and failure cases.

SELECTED PROJECTS

Brown Course Review Webapp (Java, Firebase, TypeScript, React)

Providence, RI | Fall 2024

- Built a Java (Spark) HTTP API + Firebase data layer serving reviews for 2,700+ courses and 1,600+ professors.
- Wrote and maintained unit and integration tests for the server with 75-100% test coverage and conducted end-to-end testing using Playwright to ensure high reliability.
- Collaborated with a team of 4 on a scalable solution, optimizing API queries and data processing and retrieval for a concurrency-safe and user-centric application. ([GitHub](#))

Gaussian Splatting Pipeline (Python, PyTorch)

Providence, RI | Spring 2025

- Designed and implemented a full 3D Gaussian Splatting training pipeline: defined model architecture, initialized and optimized fully differentiable parameters and trained the model end-to-end for 3D reconstruction.
- Built a mixed-precision PyTorch workflow on a High Performance Computing cluster, monitored over 58 hours of training, diagnosed and debugged numerical stability issues in differentiable rendering. ([GitHub](#))

Realtime Graphics Renderer (C++, OpenGL/GLSL)

Providence, RI | Fall 2025

- Built an end-to-end real-time rendering pipeline and collaborated on a 4-person integration phase that merged independent subsystems into a unified codebase.
- Implemented GPU instanced rendering for repeated geometry, cutting per-frame render time from 800 ms to 0.015ms on demanding scenes and stress-tested instancing on a procedurally generated scene file with repeated primitives.
- Implemented additional shading + post-processing effects including color grading, stylized filters, and multiple screen-space passes, and designed a custom LUT for the final scene

RayTracer (C++)

Providence, RI | Fall 2025

- Built a C++ ray tracer from scratch, designing the renderer architecture and implementing ray generation, intersections, shading, and recursive reflections.
- Debugged floating-point precision issues and reflection-recursion failures that produced rendering artifacts, using scene-based visualization to improve numerical robustness across complex reflective scenes.
- Implemented anti-aliasing and mipmapped texture filtering, caching mip levels and using separable kernels to improve sampling efficiency and image quality in the rendering pipeline.

Technical Skills: Python, C++, Java, OpenGL/GLSL, NumPy, TensorFlow, PyTorch, React; Tools: Adobe Lightroom, Photoshop