

# ANIRUDH SARVE

Mumbai, Maharashtra, India | +91 93225 81894 | anisarve2006@gmail.com | anirudhsarve.netlify.app  
github.com/anisarve2006 | linkedin.com/in/anirudhsarve

## PROFESSIONAL SUMMARY

Systems Software and Autonomous Robotics Engineer with deep expertise in state estimation, sensor fusion, and low-level resource-constrained edge computing. Proven success designing physics-constrained Kalman filters (15-state ESKF), compiling deep neural networks into sub-30 KB ONNX graphs for automotive ARM Cortex-A ECUs, and developing real-time operating system kernel simulators. Hands-on experience across C++, Python, Linux/POSIX, ROS, and embedded telemetry stacks.

## EDUCATION

**Vidyalankar Institute of Technology (VIT), Mumbai — B.E. in Computer Engineering**

Expected May 2028

**Cumulative GPA: 8.90 / 10.0** | Affiliated with University of Mumbai

**Relevant Coursework:** Operating Systems, Computer Architecture, Data Structures & Algorithms, Robotics & Autonomous Systems, Sensor Fusion, Linear Algebra, Probability & Random Processes, Microprocessors.

**Sonopant Dandekar Shikshan Mandali — HSC (Class XII, PCMB)**

May 2024

**Score: 92.60%** | Distinction in Mathematics, Physics & Computer Science

## TECHNICAL SKILLS

**Languages:** C++, C, Python, Rust (Basics), Java, Assembly (x86/ARM basics), Bash, SQL.

**Robotics & State Estimation:** Error-State Kalman Filter (ESKF), Extended Kalman Filter (EKF), Strapdown IMU Mechanization, Sensor Fusion (IMU, Odometry, GNSS), Non-Holonomic Constraints (NHC), ZUPT, ROS / ROS 2 (Basics).

**Embedded & Edge Runtimes:** ONNX Runtime, ARM Cortex-A ECUs, Embedded Linux, POSIX Threads (pthreads), WebAssembly (WASM), WebGL, Serial Telemetry (UART, I2C, SPI basics).

**Systems & OS Concepts:** Process Scheduling (FCFS, SJF, Round Robin, Priority), Contiguous Memory Allocation (First-Fit, Best-Fit, Worst-Fit), Memory Fragmentation, Concurrency & Mutexes, Virtual Memory.

**Tools & Platforms:** Git, Linux/Unix, Docker, GCC/Clang, GDB, Valgrind, Matplotlib TkAgg, CustomTkinter.

## SYSTEMS & ROBOTICS EXPERIENCE

**Indian Rover Challenge (IRC) — Autonomous Navigation Team — Robotics Software Engineer**

Jan 2024 – Present

- Developed real-time onboard telemetry pipelines and kinematic estimators for an autonomous planetary rover competing in the Indian Rover Challenge.
- Programmed multi-sensor serial data acquisition routines in **C++** reading raw 9-DOF IMU data, rotary encoders, and GPS fixes over serial buses with dedicated packet checksum validation.
- Implemented state-estimation filters fusing IMU angular velocity and accelerometer outputs to counteract mechanical chassis vibration and terrain slip during outdoor rough-terrain traversal.
- Mentored junior robotics members in Linux systems, cross-compilation toolchains, and sensor interfacing.

*Exposure: C++, Python, ROS, Serial Communication, IMU Sensor Fusion, Kinematics, Linux.*

**NxtChapter — Software Developer (Edge Systems & WASM)**

Nov 2024 – Aug 2025

- Compiled computer vision neural networks into client-side **WebAssembly (WASM)** and WebGL binaries, offloading heavy matrix compute directly to edge device CPUs/GPUs.
- Reduced edge segmentation latency to under **3.5 seconds**, eliminating recurring cloud GPU costs while enforcing strict memory containment inside client sandboxes.
- Built low-latency state synchronization mechanisms utilizing atomic distributed locks and WebSocket messaging under high concurrency.

*Exposure: WebAssembly, WebGL, Low-Latency Networking, Memory Sandboxing, Node.js.*

## FEATURED TECHNICAL PROJECTS

**15-State Error-State Kalman Filter (ESKF) & Neural Dead-Reckoning Engine for ARM Cortex-A ECUs (ISRO)**

**Technologies:** C++, Python, 15-State ESKF, ONNX, ARM Cortex-A

- Engineered a hybrid inertial dead-reckoning engine for ISRO (PS 26168) holding autonomous vehicle position drift to  $\leq 1.85$  m across prolonged 120-second complete GPS blackouts (a **98.4% drift reduction** versus standard EKF baselines).
- Fused 10 Hz raw IMU strapdown mechanization with a 15-state Error-State Kalman Filter ( $\delta p, \delta v, \delta \theta, \delta b_a, \delta b_g \in \mathbb{R}^{15}$ ), enforcing physical Non-Holonomic Constraints (NHC) and stationary Zero-Velocity Updates (ZUPT).
- Designed a lightweight 5-head temporal self-attention + GRU neural network predicting forward velocity and dynamic measurement covariance ( $R_v$ ), dynamically bounding drift while preventing filter divergence.
- Compiled the neural model into a **24.6 KB ONNX graph** executing in **sub-3 ms** on automotive ARM Cortex-A ECUs with zero GPU requirement and zero cloud communication.

## Multimodal Optical-SAR Generative Reconstruction Network with Boundary Invariants (ISRO LISS-IV)

*Technologies: PyTorch, ResU-Net, Cross-Modal Attention, SAR*

- Developed an operational generative AI framework for automated cloud removal and surface reconstruction in ISRO Resourcesat-2/2A LISS-IV satellite imagery (5.0 m resolution, Green/Red/NIR bands).
- Engineered a multi-encoder ResU-Net with cross-modal attention bridging cloudy optical LISS-IV observations, cloud-penetrating Sentinel-1 SAR radar (VV/VH backscatter), and multi-temporal clear-sky reflectance priors.
- Enforced hard clear-pixel preservation ( $M \cdot G + (1 - M) \cdot \text{Orig}$ ) to maintain 100% bit-exact original observations for cloud-free pixels, restricting generative synthesis strictly to occluded regions.
- Formulated a differentiable Spectral Angle Mapper (SAM) loss and aleatoric Laplacian uncertainty head, achieving **32.09 dB cloud-region PSNR**, **0.7167 SSIM**, and a tight **6.17° SAM angle** on real ISRO validation scenes over the Assam corridor.

## Real-Time Kernel Process Scheduling Simulator & Contiguous Memory Partitioning Virtual Machine

*Technologies: Python, CustomTkinter, Matplotlib TkAgg, Multiprocessing*

- Built an interactive operating system kernel simulator modeling CPU process scheduling algorithms (preemptive and cooperative FCFS, Shortest Job First, Round Robin with dynamic quantum, and Priority Scheduling) with dynamic Gantt canvas rendering.
- Implemented contiguous physical memory allocation models (First-Fit, Best-Fit, and Worst-Fit) simulating a partitioned 1024 MB address space with real-time internal/external fragmentation tracking.
- Visualized process state transitions, waiting times, and turnaround metrics under varying system loads to simulate deadlock avoidance and resource contention.

## High-Performance Vector Graphics Stream Compiler via Rust N-API Canvas Bindings

*Technologies: Rust (@napi-rs/canvas), TypeScript, Node.js*

- Built a high-performance vector rendering pipeline compiling structured document schemas directly into raw vector PDF streams in **under 150 ms** using Rust-native canvas bindings.
- Achieved an **88% memory reduction (<40 MB RAM)** compared to headless Chromium / Puppeteer processes (>350 MB RAM), preventing memory leaks and process saturation under continuous rendering loads.

## Air-Gapped Multimodal Media Understanding Pipeline with In-Memory Float32 Vector Search

*Technologies: Python, SQLite BLOB Vectors, Faster-Whisper, pHash*

- Developed a local-only multimodal media understanding engine running entirely in-memory on 16 GB RAM laptops with zero network connectivity.
- Engineered a sequential float32 BLOB vector scan in SQLite completing semantic queries across 50,000 video segments in **under 12 ms** without requiring heavyweight vector-database server daemons.
- Implemented perceptual hashing (pHash) and RGB histogram distance on raw CPU, pruning **85% of redundant visual frames** in <1 ms.

## HONORS & LEADERSHIP

- **Lead Architect, ISRO Smart India Hackathon (SIH 2026):** Developed the winning AI-ESKF sensor-fusion algorithm for problem statement PS 26168 (Smart Vehicles).
- **Autonomous Navigation Member, Indian Rover Challenge (IRC):** Represented Vidyalankar Institute of Technology in national robotics rover trials.
- **Academic Merit:** 8.90/10.0 CGPA at VIT Mumbai; 92.60% in HSC Board Examination.