SHIVAM SHARMA

4802085286| sshar443@asu.edu |github.com/shivam15112003|www.linkedin.com/in/ss1511| https://shivam15112003.github.io/shivam-portfolio/

SUMMARY

AI engineering graduate pursuing an MSc in Robotics & Autonomous Systems (AI). Skilled in ML/DL, computer vision, IoT, gesture interfaces, and cloud-scale AI systems. Actively seeking an internship in robotics software, AI/ML, computer vision, or data science.

EDUCATION

Master of Science in Robotics and Autonomous Systems (AI) May 2027
Arizona State University, Tempe, AZ
Bachelor of Technology in Artificial Intelligence Aug 2021- May 2025
Amity University, Noida, Uttar Pradesh, India 9.8/10

TECHNICAL SKILLS AND CERTIFICATIONS

Programming Languages: Python, SQL, C/C++, JavaScript

Machine Learning & Data: PyTorch, Transformers (Hugging Face), TensorFlow, scikit-learn, Pandas, NumPy, XGBoost, LightGBM, CatBoost, MLflow, Optuna, Keras, Matplotlib.

Computer Vision & Robotics: OpenCV, ROS2, ONNX, MediaPipe, dlib

NLP: Transformers (Hugging Face), spaCy, Sentence-BERT, NLTK

Certifications: Microsoft AI, Applied AI (IBM/Coursera), Aerial Robotics (University of Pennsylvania), Python for Data Science (NPTEL)

PROFESSIONAL EXPERIENCE

AI Engineer January 2025–June 2025
Salesforce, Gurgaon, Haryana, India - Tech Stack: Python, PyTorch, TensorFlow/Keras, OpenCV, ONNX, MLflow, NumPy, Pandas, scikit-learn, librosa, Matplotlib

- Shipped a **real-time multi-attribute face analytics** service (age band, **emotion**, **attire**, nationality) by fine-tuning **MobileNetV2** and exporting to **ONNX**, enabling **30 FPS** with **<30 ms** per-frame latency and delivering **94–95% macro-F1** on a held-out set.
- Built a **driver drowsiness & distraction** system by fusing **blink-rate**, **PnP head-pose**, and a **CNN yawning detector**, achieving **0.92 F1** on **20+ hours** of **dash-cam video** and improving **safety** via **real-time alerting**.
- Developed a **multimodal emotion classifier** combining a **facial CNN** with a **BiLSTM** over **MFCCs** (female voice), reaching **91–94% accuracy** and accelerating iteration via automated labeling/augmentation (**SpecAugment**, **mixup**) with experiment tracking in **MLflow**.

Data Scientist Intern April 2024–June 2024
Tata Consultancy Services, Noida, Uttar Pradesh ,India -Tech Stack: Python, scikit-learn, Optuna, SMOTE (imblearn), Pandas, NumPy, SHAP, Streamlit, Matplotlib

- Built an **early-warning churn score** for **500K+ customers** using **real-world signals**; **~25% improvement** over the prior approach at identifying likely churners.
- Built a **reusable scikit-learn pipeline** with **target encoding**, **SMOTE**, **time-aware cross-validation**, and **Optuna hyperparameter search**; produced a **model card** with **stability/fairness checks**.
- Explained drivers with **SHAP** and delivered a lightweight **Streamlit** dashboard for Ops; **reduced false positives by 18%** at fixed recall in **back tests**.

ACADEMIC PROJECTS

Agentic Robot Control via LLM/VLM (Prompt-to-Action) Sep 2025 – Dec 2025

- Built **agentic AI** pipeline turning **natural-language prompts** into parameterized **pick/place/rotate** skills (e.g., "pick the small blue block, rotate 90 deg, place on red block"); expanded **prompt templates**. *Tools/Languages:* **Python**, **PyTorch**, **OpenCV**, **ROS 2/ROS2 (rclpy/rclcpp)**, **inverse kinematics (IK)**, **gripper control**.
- Added **monocular depth estimation** for **z-aware scene understanding** and **kinematic planning**; composed **perception -> planning -> execution** with **safety checks** and **recovery** using **tf2** and **ROS 2 nodes**.
- Demonstrated precise **grasp/placement** across varied **size/color/rotation** constraints; instrumented runs with **rosbag2** and **ros2 launch**.

Dobot Magician: Agentic Tic-Tac-Toe (Vision + LLM Planning) Aug 2025 – Sep 2025

- Built **computer vision board-state detection**: **perspective correction**, **color/edge segmentation**, **AprilTag** corners, **camera calibration**; commanded **Dobot Magician** via **ROS 2/ROS2** for precise **X/O placement**. *Tools/Languages:* **OpenCV**, **AprilTag**, **ROS 2 (rclcpp)**, **tf2**, **Python**, **C++**.
- Orchestrated **perception -> planning -> actuation** with **Gemini LLM** via **function calls** (perceive_board, choose_move — **Minimax** + **alpha-beta**, execute_move); added **IK limits**, **safety bounds**, **robust recovery** for **illegal/ambiguous states**. *Tools:* **ros2 launch**, **rosbag2**, **ros2_tracing**.
- Achieved **~1.4 s p50 latency** and **<= 2 mm placement error** over **200 games**; **profiling** and **logs** validated stability.

ROS2 Gesture-to-Robot: Vision-based Tele-operation for Mobile Robots Jan 2025 – Apr 2025

- Implemented **real-time hand/pose interface** mapping **gestures** to **TurtleBot navigation** and **gripper** actions; **end-to-end latency ~55 ms**. *Tools/Languages:* **MediaPipe**, **OpenCV**, **ROS 2/ROS2 (Python/C++)**, **Gazebo**.
- Reached **>= 95% F1** on **custom gesture dataset** with **2.8 cm mean path error** in **simulation**; added **safety gestures** and **low-pass filtering** to reduce **jitter**.
- Delivered **>= 97% gesture-to-action reliability** and **<= 120 ms safe-stop** via **ROS 2 safety supervisor** (**debounce**, **Kalman smoothing**, **dead-man open-palm**), **BehaviorTree.CPP** gating of **cmd_vel/gripper**, and **QoS tuning** (reliable, sensor_data).