Mohamed **Souab**

M2 Systèmes automatiques mobiles · Stage de fin d'études (PFE) 2027

Paris, France

+33 7 73 18 98 79 | mohamed.souab2@gmail.com | mosouab.github.io | mosouab | mosouab



Formation

M2 Systèmes automatiques mobiles (E3A), majeure Perception · Université Paris-Saclay — *2026 – 2027*

M1 Électronique, énergie électrique et automatique (E3A) · Université Paris-Saclay — *2025 – 2026*

Licence Électronique, automatique et traitement de l'information · Université d'Évry Paris-Saclay — *2024 – 2025*

DUT Informatique industrielle et systèmes automatisés · EST de Kénitra, Maroc — *2022 – 2024*

Expérience professionnelle

Groupe Bel

Stagiaire ingénieur en automatisme — *Avril 2024 – Juin 2024*

- Migration des automates Siemens S7-300 vers S7-1516 : le diagnostic avancé du nouveau CPU a réduit les micro-arrêts et fait passer le TRS de **74 % à 88 %**.
- Conception et mise en service d'un suivi temps réel des KPI de maintenance (TRS, BKD, MTBF, MTTR), avec archivage centralisé des alarmes sur une IHM Unified MTP700 développée en HTML, CSS et JavaScript. Le MTTR est descendu de **45 à 12 minutes**.
- Script Python pour le mappage des adresses de bits d'alarme, jusque-là saisies à la main : **10 minutes au lieu de 3 jours** de travail.
- Passage de Profibus DP à PROFINET sur le S7-1516, avec un temps de cycle des E/S ramené de **12 ms à 2 ms**.
- La boucle de retour capteur, plus rapide sur le nouveau CPU, a fait baisser le déclassement du film aluminium (tircel) de **3,5 % à 1,1 %**.

Projets

Drone autonome sans GPS : SLAM visuel et atterrissage de précision

github.com/mosouab/drone_px4_slam

Projet personnel

- **Architecture :** Raspberry Pi 4 sous ROS 2 (Jazzy), reliée au Pixhawk PX4 par MAVROS, sur châssis Holybro X500 V2.
- **Localisation :** odométrie visuelle et cartographie 3D (SLAM RTAB-Map) avec une caméra de profondeur Intel RealSense D455, fusionnées dans l'EKF2 de PX4.
- **Atterrissage de précision :** marqueurs ArUco (détection OpenCV) et asservissement en vitesse.
- **Missions autonomes :** décollage, waypoints et RTL via les API constructeur (MAVLink en C, MAVSDK en Python), testées sous Gazebo en SITL avant le vol réel.
- **Déploiement :** conteneurisation de la pile ROS 2 sous Linux et réduction de la charge CPU sur la Raspberry Pi pour les essais en vol.

EdgeSense : maintenance prédictive embarquée pour machines industrielles

github.com/mosouab/EdgeSense

Programme Innov'Action (Paris-Saclay) · **Sélectionné pour une learning expedition au MIT (Boston)** — *Mai 2026 – aujourd'hui*

- **Approche :** détection d'anomalies non supervisée. Le modèle apprend le fonctionnement normal d'une machine sur quelques semaines de données saines, sans historique de pannes étiqueté.
- **Modèle :** autoencodeur convolutif 1D de type USAD en PyTorch (41,5 k paramètres, 0,17 Mo). Il donne un score d'anomalie et un indice de santé de 0 à 100 % qu'un technicien lit directement.
- **Résultats :** sur les compresseurs du métro de Porto, les 4 pannes documentées ont été signalées dès leur apparition ; le modèle a aussi été testé sur un banc hydraulique à 4 composants.
- **Benchmark :** sur TSB-AD (35 séries industrielles), 1er devant 23 méthodes publiées sur le jeu CATSv2 (VUS-PR).
- **Inférence en périphérie :** 0,5 ms par fenêtre sur un seul cœur CPU ; les données brutes ne quittent pas le site.
- **Exploitation :**
  - simulateur d'équipement edge en temps réel (FastAPI, WebSocket, bus de messages de type MQTT) ;
  - boucle de retour opérateur (faux positif ou panne confirmée) qui adapte le modèle sur site.

Compétences

Contrôle	Représentation d'état, commande par retour d'état, observateurs, asservissement visuel et commande référencée capteurs, automates Siemens (TIA Portal, PROFINET)
Perception	SLAM, stéréovision, planification de trajectoire et évitement d'obstacles, fusion multicapteurs (EKF/UKF, filtres particulaires), traitement du signal et de l'image, OpenCV
Développement	C++, C, Python, ROS 2, FreeRTOS, STM32, VHDL (FPGA), bus CAN, I2C/SPI/UART
Simulation & test	MATLAB/Simulink, Gazebo, SITL, jumeaux numériques, validation sur système réel
Apprentissage automatique	Apprentissage supervisé et non supervisé, CNN (détection, segmentation), autoencodeurs, optimisation, PyTorch, scikit-learn
Linux & déploiement	Linux, Docker, Git, Bash, administration de serveurs
Langues	Français C2 (TCF) · Anglais C1 (TOEIC 920/990)