CV/ML Software Engineer

Experience: 2–3 years
Employment Type: Full time

About the Role

We’re building safety first video telematics products (ADAS/DMS/driver behavior analytics) that run efficiently on edge devices inside commercial vehicles. You will write modern C++ software, integrate and optimize CV/ML pipelines, and ship reliable, low latency perception features such as driver monitoring and distance estimation from camera feeds.

What You’ll Do

- Own C++ software modules for on device video capture, preprocessing, inference, and post processing on Linux.
- Implement classical image processing pipelines (denoise, resize, color space, undistortion) and CV algorithms (keypoints, homography, optical flow, tracking).
- Build and optimize distance/spacing estimation from monocular/stereo camera(s) using calibration, geometry, and/or depth-estimation networks.
- Integrate ML models (PyTorch/TensorFlow → ONNX/TensorRT/NNAPI/NPU runtimes) for DMS/ADAS events: drowsiness, distraction/gaze, phone-usage, smoking, seat belt, etc.
- Hit real time targets (FPS/latency/memory) on CPU/GPU/NPU using SIMD/NEON, multithreading, zero copy buffers.
- Write clean, testable C++, CMake builds, and Git based workflows (branching, PRs, code reviews, CI).
- Instrument logging/telemetry; debug with gdb/addr2line, sanitize and profile with perf/valgrind.
- Collaborate with data/ML teams on dataset curation, labeling specs, training/evaluation, and model handoff.

· Work with product & compliance to meet on road reliability, privacy, and regulatory expectations.

Must Have Qualifications

· 2–3 years hands on in **modern C++** on Linux, with strong **Git** and **CMake**.

· Solid **image processing** and **computer-vision** foundations (camera models, intrinsics/extrinsics, distortion, PnP, epipolar geometry).

· Practical experience integrating **CV/ML models** on device (OpenCV + ONNX Runtime/TensorRT/NCNN/MediaPipe/NNAPI).

· Experience building **real time pipelines** for live video (GStreamer/FFmpeg, RTSP/RTMP, ring buffers), optimizing for **latency & memory**.

· Competence in **multithreading/concurrency**, lock free queues, and producer–consumer designs.

· Comfort with **debugging & profiling** on Linux targets.

Good to Have (Nice Bonuses)

· Knowledge of monocular depth estimation, stereo matching, or structure from motion for **distance estimation**.

· Model training exposure (**PyTorch/TensorFlow**): augmentation, evaluation (precision/recall, ROC/PR), quantization/pruning, conversion to ONNX/TensorRT/NCNN.

· Hardware acceleration (GPU/VPU/NPU, Arm **NEON**/DSP), YOLO/RT DETR/Lightweight backbones on edge.

· Cross compiling, Yocto/Buildroot, containerized toolchains; unit tests (gtest), static analysis (clang tidy, cppcheck), sanitizers.

· Basic familiarity with **MQTT/IoT**, message schemas, and over the air updates.

Tools You'll Use

- **Languages:** C++, Python
- **CV/ML:** OpenCV, ONNX Runtime/TensorRT/NCNN/MediaPipe; PyTorch/TensorFlow (for training/eval).
- **Video:** GStreamer/FFmpeg, V4L2, RTSP/RTMP.
- **Build/DevOps:** CMake, Git, gtest, clang-tidy, sanitizers; CI/CD (GitHub/GitLab/Bitbucket).
- **Debug/Perf:** gdb, perf, valgrind

Qualifications

- B.Tech/B.E. in CS/EE/ECE (or equivalent practical experience).
- 2–3 years in CV/ML or video-centric software roles with a Robotics background.