

R. Raphael Del Popolo

rrdelbuffalo@icloud.com 623.274.5266 Interim Secret Clearance BA + AAS

SOFTWARE ENGINEER – MACHINE LEARNING & GENERATIVE AI

Software Engineer focused on LLM and Computer Vision evaluation systems, data preparation and production delivery. At Meta I built the LLM classifiers and judges behind an internal coding assistant, defined the failure-mode taxonomy and built the triage guide. Since then I have shipped on-device Vision pipelines with LLM integration – owning the data preparation behind them, 61,000 labeled pose frames now auto-labeled by a vision-language model, and the optimization that keeps every model on-device.

TECHNICAL SKILLS

GENERATIVE AI	LLM classification, grading and evaluation harnesses · retrieval/embedding pipelines and LoRA fine-tuning · vision-language models (Qwen3-VL) · on-device tool calling · data preparation and optimization
DATA & EVALUATION	Dataset design, versioning and review tooling · held-out bakeoffs against an established noise floor · leakage audits · SQL dashboards over large evaluation runs
COMPUTER VISION	Detection, segmentation (SAM 2/3), keypoints, pose and tracking · auto-labeling · PyTorch → CoreML / ONNX on the Neural Engine, FP16, latency-profiled
STACK	Python · PyTorch · Swift 6 · TypeScript · SQL · OpenCV · CoreML · ONNX Runtime · SQLite · Git · Unix/Linux CLI

EXPERIENCE

Lead Software Engineer, Computer Vision UTR Sports · Remote 2026 – Present

- Sole engineer building GenAI-powered coaching features on top of an end-to-end vision pipeline that turns handheld phone video — outdoors, uncontrolled lighting, arbitrary camera placement — into measured biomechanics, delivered across three macOS apps, an iOS app, and a watchOS companion.
- Built a natural-language and voice command layer through Apple Foundation Models. Took advantage of Apple Foundation Models such as Vision (Pose, Segmentation, OCR), Sound Analysis for maximal optimization of local hardware.
- Hand-labeled the corpus, then cut that work down with a Qwen3-VL → SAM 3 auto-labeler that beat the hand labels on recall — a first pass a human still reviews, not an unsupervised one. Owned the full data-preparation pipeline behind 61,000 labeled pose frames.
- Exported every model PyTorch → CoreML/ONNX on the Apple Neural Engine, all compute on-device with no server round-trip, later ported to a C++ web build to reach a second platform from one codebase.
- Raised coaching-cue coverage from 30% to 81% under a reproducible evaluation workflow: held-out sets, baseline comparisons, leakage checks, explicit failure tracking. Used professional players to calibrate the model.

Software Engineer (Contractor) Meta via Wipro · Remote 2025 – 2026

- Worked on Meta's internal LLM for software development — the assistant engineers used inside VS Code — evaluating several product surfaces against a corpus in the low thousands, so a finding had to hold across more than one way engineers actually reached the model.
- Improved an AI coding assistant's measured accuracy by 80% through controlled prompt and context experiments evaluated against a fixed test set; defined the failure-mode taxonomy used to count and classify its errors.
- Designed classification systems for thousands of LLM outputs and built the SQL dashboards over them, cutting analysis time 90%.
- Wrote the case-handling guides the triage team worked from, one documented path per class of case, and kept the running record of recurring problems that went back to the engineers who owned them — so reviewers could diagnose model failure modes without escalating at all.
- Triaged over 1,000 cases as the escalation path for front-line reviewers, 100+ issues in a week at peak.

Software Engineer Operation Care and Comfort · Remote 2023 – 2024

- Built the platform that matched donated goods to the families who qualified for them, with eligibility driven by household profile, geolocation, and reward history.
- Automated the hand-out: until then every donated event ticket was matched to a family by hand, so I wrote the eligibility rules and the allocation run that now releases hundreds of tickets at once. Uncountable hours a week regained.

Local Model Environment – Self-Hosted LLM Pipelines

Python

Qwen3

DeepSeek-R1

Command-R

Gemma

ComfyUI

Wan 2.2

LoRA

SQLite

- Open-weight models run entirely on my own hardware, serving thousands of batch requests from Python and local coding agents.
- Used as data-correction workers over databases too large to hand-check: a USDA food table passed row by row through a local model to correct mis-typed entries and derive new fields from deduction (recipe-> type of meal), resumable and written after every row; a separate 3,760-item notes and bookmark corpus tagged, embedded and indexed for search.
- Image generation and editing. Local diffusion models – FLUX.1, Illustrious XL (SDXL) and Qwen-Image-Edit across Draw Things and ComfyUI – including LORA training over ComfyUI's local HTTP API.
- Video generation. Wan 2.2 and MiniMax image-to-video locally.

Autonomous Drone Navigation – DJI Tello

Python

PyTorch

OpenCV

SLAM

- Closed-loop visual autonomy on a DJI Tello airframe: H.264 stream decoded over UDP with OpenCV, perception per frame, RC velocity commands back at 10 Hz, no external positioning of any kind.
- Single-camera obstacle avoidance from MiDaS DPT-Large depth with a pitch-shifted collision-corridor ROI; made relative depth metric by fitting a pixel-intensity → distance model from measured standoff flights. Nominated for the Intel AI Global Impact Award.

Punchline – Self-Supervised Video Reel Cutter

Swift 6

SoundAnalysis

AVFoundation

SQLite

- Dissects stand-up specials and cuts vertical social reels from the biggest laughs. The audience laugh is the label, so the system is self-supervised: audio drives detection and ranking, video only refines where to cut.

Ranger Sandbox – Tactical Training & Simulation

Python

JavaScript

React

SVG

- Eight offline training stations to train soldiers on perishable skills and otherwise dry doctrine.
 - A scenario generator covering terrain, enemy, troops and time; contradiction detection names conflicting inputs, and 1,000 saved scenarios are re-validated after every change. Map import calibrates a screenshot to true scale from two known points, then OpenStreetMap and aerial imagery estimate terrain-crossing difficulty and movement time.
- Scenario Builder – fire-team scenarios, exports a card
 - Nine-Line Radio – emergency call scored on procedure
 - Drill + AAR – decide under fog, read the cited verdict
 - Doctrine Deck – deck built from scanned pages

- Sand Table – squad formation graded on spacing and security
 - Atoms Trainer – drill steps as spaced repetition
 - Map Import – screenshot to terrain at true scale
 - Procedure Trainer – animated M4A1 field strip

EDUCATION, CERTIFICATIONS & ADDITIONAL

A.A.S. Artificial Intelligence & Machine Learning · A.A.S. Computer Science – Chandler-Gilbert Community College	U.S. National Guard, Infantry – squad leadership, live-fire operations; certified in first aid and combatives.
B.A. Acting & Film – University of Central Lancashire (UCLan)	Photography & Videography – optics, exposure and sensor behaviour outdoors, advanced camera systems.
FEMA Certified + Interim Secret clearance	Languages – German fluent · Spanish fluent · French conversational.