# Deterministic and Auditable On-Device Polynomial Inference for On-Board SAR Oil-Spill Screening

Vahram A. Martirosyan*
*Dept. of Computer and Information Science*
*Higher Colleges of Technology*
Abu Dhabi, United Arab Emirates
vmartirosyan@hct.ac.ae

***Abstract*—Oil spills must be found quickly, yet SAR imagery is today downlinked in full and analysed on the ground. Running a spill detector aboard the satellite—on a radiation-tolerant microcontroller-class core—would cut downlink and latency, but whether a useful detector fits such a budget is unproven. This paper is a feasibility study, not an operational detector. A frozen principal-component (PCA) front end feeds a closed-form degree-2 polynomial (SRPR) head; the whole model is folded and quantised into a fixed-point kernel. On the space-borne Deep-SAR SOS benchmark (Sentinel-1 C-band and ALOS PALSAR L-band; 54,371-tile held-out set) it reaches G-mean 0.822. We quantify the feasibility envelope: int8 storage shrinks the model from 657 KB to 82 KB at a G-mean cost of 0.001; a resolution/rank ladder reaches a 15 KB model at G-mean 0.803, which fits an ATmega328P. On four real microcontrollers the integer kernel runs in 3.76–9.2 ms/tile (32-bit) and 83.6 ms (ATmega328P); instruction-accurate simulation gives 1.3–1.9 ms on a rad-tolerant Cortex-M7. We also delimit the claim: tile presence, not segmentation (coarse-mask IoU 0.38); radiometric tolerance $\sim \pm 1.5$ dB; look-alike false alarms unvalidated; and a baseline study in which tree ensembles (XGBoost 0.884, random forest 0.870) and an int8 MLP (0.870) are more accurate but larger or opaque. The polynomial head is the only candidate that is simultaneously microcontroller-deployable, deterministic (convex, reproducible training; input-independent integer inference) and auditable term-by-term. The contribution is a quantified feasibility envelope and an honest baseline.**

***Index Terms*—marine electronic information, edge AI, TinyML, synthetic aperture radar, oil-spill detection, interpretable machine learning, polynomial regression, resource-constrained embedded systems.**

## I. Introduction

Marine oil spills demand rapid detection to bound ecological and economic damage. C-band SAR is the operational sensor of choice (e.g., EMSA CleanSeaNet) because it images through cloud and darkness, and a slick appears dark against the speckled sea [1], [3]. A growing fraction of Earth-observation satellites add on-board compute so imagery is filtered or analysed in orbit, relieving the downlink [4], [5], yet high-accuracy oil-spill models remain segmentation networks that assume a ground GPU [11]–[13]. On-board deep models either use edge accelerators or incur hundreds of milliseconds per tile [8], [10].

We ask a deliberately narrow feasibility question: *can a tile-level oil-spill classifier run on a radiation-tolerant microcontroller-class core, at low-millisecond latency and within tens of kilobytes, while retaining useful accuracy?* We instantiate it as a Stagewise-Regularized Polynomial Regression (SRPR) head on a frozen PCA front end and treat the embedded budget as a first-class design goal. We do not claim operational readiness. Our contributions are:

1) **Feasibility envelope.** Normalisation is folded into the basis and the PCA projection is quantised to int16/int8 (657 KB $\rightarrow$ 82 KB, $\Delta$G-mean $= -0.001$; Table 5); a resolution/rank ladder reaches a 15 KB ATmega328P model (Table 6).
2) **Real-hardware evidence.** An allocation-free kernel is measured on four microcontrollers and instruction-counted on Cortex-M class cores, with a clean desktop-versus-embedded separation (Tables 9,10).
3) **Honest baseline and iso-footprint analysis.** Threshold, logistic, kNN, SVM, random forest, MLP and XGBoost are compared on identical data and features (Tables 7,8); the polynomial head is deployable and auditable but not the accuracy leader, and an int8 PCA+MLP is the strongest *deployable* alternative.
4) **Determinism and auditability.** Convex closed-form training is reproducible; the integer inference has an input-independent operation count (fixed worst-case time); and the decision is an explicit, term-by-term auditable polynomial (Fig. 3).

The paper is organised as follows. Section II reviews related work; III the data and validation protocol; IV the model and its fixed-point form; V accuracy and ablations; VI baselines and the deployment envelope; VII embedded measurements; VIII robustness; IX determinism and auditability; X discusses limitations; XI concludes.

*Corresponding author.

Table 1
DATA USED (SPACEBORNE SAR). THE STOCK TRAINING SPLIT IS PALSAR-ONLY; THE BALANCED SPLIT DRAWS EQUALLY FROM BOTH SENSORS.

| Split | Sensor(s) | Scenes | Tiles | Role |
|---|---|---|---|---|
| train (stock) | PALSAR | 3,101 | 60,000 | fit (%) |
| train (balanced) | PALSAR + Sentinel-1 | 6,455 | 60,000 | fit |
| validation | PALSAR + Sentinel-1 | 1,615 | 54,371 | held-out |

## II. RELATED WORK

### A. SAR oil-spill detection

The refined Deep-SAR oil-spill (SOS) dataset [1], [2] provides Sentinel-1 and ALOS-PALSAR chips with binary oil masks. Most methods are segmentation networks—FC-DenseNet, DeepLabV3+/SegNet, SAM variants [11]–[13]—trained on ground infrastructure and not written for an on-board compute budget.

### B. Orbital edge computing and on-board AI

Orbital edge computing processes data in orbit to relieve the downlink [4]–[7]. On-board demonstrations include MobileSAM on an Unibap iX10-100 [8], a 7 W multi-task payload [9], and WISE on the ISS-mounted IMAGIN-e payload at 0.10–0.26 s/tile [10]: these use edge modules or seconds-scale latency. We target microcontroller-class silicon and milliseconds.

### C. TinyML, quantization and novelty

MicroNets/MCUNet show CNNs fit microcontrollers [14]; surveys and studies characterise int8/int16 quantization on MCU hardware [15], [16]. The same SRPR engine has been deployed on-device for multi-variable ocean profiling [17]. Sparse and symbolic regression yield compact, interpretable models but have not been evaluated for microcontroller image inference. *Novelty.* Rather than a new sparsity algorithm, our contribution is a deployment-oriented design: a frozen PCA front end whose memory dominates the model, an affine-folded projection, an integer fixed-point kernel with input-independent work, and an explicit degree-2 head whose per-term coefficients are auditable—evaluated end-to-end on real microcontrollers against iso-footprint baselines. The design choices (rank, degree, rescale, quantization) are ablated in Section V.

## III. DATA AND VALIDATION PROTOCOL

We use the refined Deep-SAR SOS dataset [1], [2], built from *spaceborne* SAR: ESA Sentinel-1 (C-band) and JAXA ALOS PALSAR (L-band). The chips are $256\times256$ single-band intensity crops (grayscale, stored as 8-bit RGB) with binary oil masks. We cut non-overlapping $40 \times 40$ tiles and label each *oil* (mask fraction $\geq 0.05$) or *clean* ($\leq 0.001$), skipping ambiguous tiles, following [1]. Table 1 summarises the data.

**Leakage controls.** The train/validation split is *scene-grouped*: the two sets contain disjoint source scenes, so tiles from one acquisition never appear in both. Pixel statistics, PCA basis, z-score parameters, and the head are fit on training folds only; the validation set is never subsampled. We additionally report a *sensor-balanced* retraining that draws equal tiles per class from both sensors (Table 3), because the stock split is PALSAR-only: cap-by-sorted-scene fills the quota before any Sentinel-1 scene is reached, so prior "Sentinel-1" numbers were cross-sensor transfer and the model is never trained on Sentinel-1.

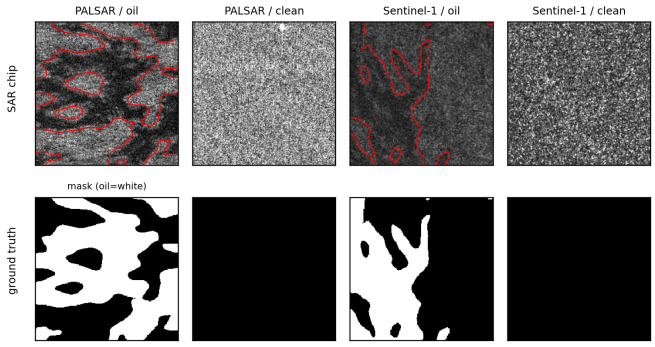



Figure 1. Representative held-out chips and ground-truth masks: positive (oil) and negative (clean) for both sensors. Images are single-band intensity SAR (grayscale replicated over RGB); red contours mark the oil masks.

Table 2
SCENE-GROUPED 5-FOLD CV (POOLED BALANCED-TRAIN + VAL TILES).

| fold | G-mean | sens | spec | PALSAR | Sentinel-1 |
|---|---|---|---|---|---|
| 0 | 0.8300 | 0.8127 | 0.8476 | 0.8488 | 0.8006 |
| 1 | 0.8300 | 0.8007 | 0.8604 | 0.8440 | 0.8077 |
| 2 | 0.8336 | 0.8134 | 0.8544 | 0.8519 | 0.8105 |
| 3 | 0.8331 | 0.7964 | 0.8715 | 0.8428 | 0.8149 |
| 4 | 0.8322 | 0.8344 | 0.8300 | 0.8623 | 0.7942 |
| mean±std | **.832±.002** | .812±.013 | .853±.014 | .850±.007 | .806±.007 |

Fig. 1 shows positive (oil) and negative (clean) chips for both sensors: the oil signature is a dark, wave-damped region and the masks are co-registered — across mixed scenes the masked oil is darker than the surrounding sea in 100% of cases (mean intensity 76 vs. 138 for PALSAR, 61 vs. 95 for Sentinel-1), confirming alignment.

**Scope of the imagery.** The chips are *processed* products (calibrated, geocoded intensity), single-polarisation (no VV/VH), matching the post-focusing product an on-board classifier consumes; SAR focusing, dual-pol exploitation and a specific $\sigma^0$(dB) normalisation are out of scope. The benchmark is curated (oil-dominant chips, negatives from oil-scene background, no explicit look-alikes) and does not by itself represent the operational scene distribution.

**Repeated evaluation.** To go beyond a single split we pool *every* labelled tile (the sensor-balanced train set plus the held-out val set; 114,371 tiles, 3,373 scenes) and run **scene-grouped 5-fold CV**, so each scene is tested exactly once with no scene leakage. The G-mean is $0.832 \pm 0.002$ (PALSAR $0.850 \pm 0.007$, Sentinel-1 $0.806 \pm 0.007$; Table 2) — stable across folds, and slightly above the fixed-split 0.822/0.826 because each fold trains on ∼91k tiles.

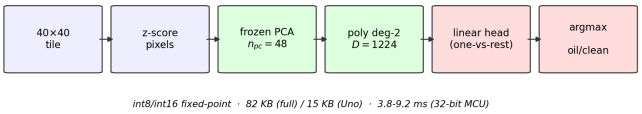



Figure 2. A $40 \times 40$ tile is z-scored, projected by a frozen PCA, expanded to degree-2 monomials, scored by a one-vs-rest linear head and assigned by argmax; the deployed kernel uses fixed-point arithmetic at 82 KB (full) / 15 KB (small).

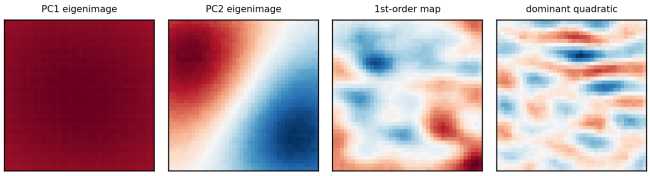



Figure 3. Auditability of the deployed model: leading PCA eigenimages, the first-order pixel-sensitivity map, and the dominant quadratic mode. The decision is a small, readable set of monomials.

## IV. Method

Each $40 \times 40$ grayscale tile is scaled to $[0, 1]$, z-scored with training statistics, projected by a frozen PCA to $n_{pc}$ components, z-scored again, and expanded to a degree-2 monomial basis ($D = 1224$ at $n_{pc} = 48$). A one-vs-rest linear head (two classes) is solved in closed form; the class is the argmax of the two scores.

*a) Normalisation folding.:* Because the pixel z-score is affine it folds into the PCA basis: with $x \in [0, 1]^P$ ($P = 1600$),

$$\mathrm{pca}_j = \textstyle\sum_d x_d A_{jd} + c_j, \quad A_{jd} = \frac{Q_{jd}}{\sigma_d}, \quad c_j = -\left(\frac{\mu}{\sigma} + \bar{z}\right)^{\top} Q_j, \tag{1}$$

removing a pass over the pixels.

*b) Fixed-point form.:* The tile is stored as $q_d = \mathrm{round}(255\, x_d)$ and the projection accumulates in int32, $\mathrm{pca}_j = (\sum_d q_d \hat{A}_{jd})\, s_A/255 + c_j$, with $|\hat{A}| \le 2047$ (int16) or $\le 127$ (int8) chosen so the accumulation cannot overflow. The memory-dominant PCA basis is integer while the small head and the per-sample $[0, 2]$ rescale stay in float32 on 32-bit cores; the head can be quantised identically.

## V. Accuracy and Ablations

The stock model reaches G-mean 0.822 on the full 54,371-tile validation set, with a large sensor gap (PALSAR 0.861 vs Sentinel-1 0.762). Retraining on the sensor-balanced split lifts Sentinel-1 to 0.795 and the overall G-mean to 0.826 at a small PALSAR cost (0.852) — a better-balanced operating point and markedly better cross-sensor generalisation (Table 3).

**Ablations.** Table 4 isolates the design choices under the same protocol. *PCA rank:* G-mean peaks at 64 components (0.832) and is flat by 96 (0.825). *Degree:* degree-3 does not help (0.75–0.79 at small rank) — degree-2 is retained. *Rescale:* the per-sample $[0, 2]$ rescale is essential (0.826 vs. 0.761 without), because it normalises each tile to its own dynamic range. *Quantization:* int16 is lossless and int8 costs 0.001 G-mean (Table 5). *Sparsification:* pruning the PCA basis degrades quickly (top-50% already $-0.011$): its energy is dense, so the win is quantization, not pruning.

Table 3
Effect of sensor-balanced training (same mixed 54,371-tile val).

| metric | PALSAR-only | balanced |
|---|---|---|
| accuracy | 0.8225 | 0.8263 |
| sensitivity | 0.8419 | 0.7848 |
| specificity | 0.8022 | 0.8699 |
| G-mean | 0.8218 | **0.8263** |
| precision | 0.8170 | 0.8636 |
| G-mean, PALSAR | 0.8609 | 0.8517 |
| G-mean, Sentinel-1 | 0.7622 | **0.7953** |

Table 4
Ablations (sensor-balanced train; full val).

| | | | |
|---|---|---|---|
| **(a) PCA rank $n_{pc}$ (deg 2, rescale on)** | | | |
| $n_{pc}$ | 48 | **64** | 96 |
| G-mean | 0.826 | **0.832** | 0.825 |
| **(b) Degree and per-sample rescale** ($n_{pc} = 48$) | | | |
| setting | deg-2 rescale | deg-2 no-rescale | deg-3 ($n_{pc}$=24) |
| G-mean | **0.826** | 0.761 | 0.793 |
| **(c) Quantization and PCA-basis sparsification** | | | |
| setting | float64 | int16 | int8 |
| G-mean | 0.8218 | **0.8218** | 0.8210 |
| footprint | 657 KB | 164 KB | 82 KB |

## VI. Baselines, Iso-Footprint and Deployment Envelope

Table 7 compares against standard baselines trained on the same sensor-balanced split and the same frozen-PCA features. Threshold, logistic and kNN are weak (G-mean 0.68–0.70). Nonlinear models are stronger: SVM-RBF 0.858, random forest 0.870, an MLP (48–64–32) 0.870, and XGBoost 0.884. The polynomial head reaches 0.826 (best 0.832 at 64 PCs).

**Iso-footprint analysis.** All methods share the same frozen PCA front end, so its integer basis dominates memory (75 KB int8); the heads differ. At matched footprint the deployable alternatives are the polynomial head (2,448 coefficients; 2.4 KB int8) and a compact int8 MLP ($48\text{–}64\text{–}32 = 5{,}282$ parameters; 5.3 KB). The MLP is *more accurate* (0.870 vs. 0.826) at a similar footprint, while tree ensembles and SVM are larger, non-integer and branch-heavy. We report this as a deliberate boundary result: on this task an int8 MLP is competitive per kilobyte, so the polynomial head's case rests on *determinism and auditability*, not on accuracy or size.

## VII. Embedded Deployment and Efficiency

The inference kernel is allocation-free and compiles for all four boards; we measure it end-to-end per tile (Table 9). On the ATmega328P the $20 \times 20/32$ int8 model uses 75% of flash and 49% of the 2 KB SRAM; the full model fits RP2040 and ESP32. The integer path gives a 14× speedup on the FPU-less RP2040, 5.6× on the ATmega328P, and 1.3× on the FPU-equipped ESP32/RP2350. Device scores match the host reference to 4–5 decimals (the host numbers below are clearly separated from the embedded ones). Keeping the PCA basis in

Table 5
STORAGE/ACCURACY TRADE-OFF (FULL 54,371-TILE VAL).

| variant | G-mean | footprint |
|---|---|---|
| float64 baseline | 0.8218 | 657 KB |
| int16 all | 0.8218 | 164 KB |
| int8 all | 0.8210 | 82 KB |
| int8 $A$ + int16 $W$ | **0.8219** | 83 KB |
| folded + int8 | 0.8215 | 82 KB |
| PCA basis top-75% | 0.8224 | 120 KB |
| PCA basis top-50% | 0.8105 | 80 KB |

Table 6
RESOLUTION/RANK LADDER (SAME RECIPE; FULL VAL).

| tile | px | $n_{pc}$ | $D$ | G-mean | int8 KB |
|---|---|---|---|---|---|
| $40 \times 40$ | 1600 | 48 | 1224 | 0.8295 | 82.2 |
| $20 \times 20$ | 400 | 32 | 560 | 0.8030 | 14.8 |
| $20 \times 20$ | 400 | 16 | 152 | 0.7314 | 7.8 |
| $10 \times 10$ | 100 | 16 | 152 | 0.7384 | 2.2 |
| $8 \times 8$ | 64 | 8 | 44 | 0.6719 | 0.8 |

RAM did not change latency: the int16 multiply–accumulate loop, not flash reading, is the bottleneck.

Table 9
MEASURED END-TO-END INFERENCE PER TILE ON REAL MICROCONTROLLERS. THE ATMEGA328P RUNS THE $20 \times 20$/32 INT8 MODEL; THE 32-BIT BOARDS RUN THE $40 \times 40$/48 INT16 MODEL.

| board (core, clock) | flash | float32 | int16 |
|---|---|---|---|
| ATmega328P ( 16 MHz) | 24.6 KB | 465 ms | 83.6 ms |
| RP2040 M0+ (133 MHz) | 222 KB | 131 ms | **9.18 ms** |
| RP2350 M33 (150 MHz) | 219 KB | 6.33 ms | **3.76 ms** |
| ESP32 LX6 (240 MHz) | 437 KB | 11.7 ms | **9.12 ms** |

**Energy.** These boards expose no on-chip power monitor, so we give first-order estimates $E = VIt$ with typical active current: $\sim$0.4–6 mJ per tile (Uno 6.3 mJ; RP2040 0.9 mJ; RP2350 0.4 mJ; ESP32 4.6 mJ) — negligible next to the SAR imager/focuser and downlink budget, so on-board inference is not the energy bottleneck. (These are estimates, not measurements.)

**Flight-hardware projection.** A functional instruction-set simulator counts retired instructions per inference (Table 10). On Cortex-M4 the full model takes 784,969 (float32) or 401,063 (int16) instructions; on the FPU-less M0 the float path is soft-float-bound (5.6 M) whereas int16 needs only 470k. Projected to radiation-tolerant cores, the full detector runs at **1.3–1.9 ms** on a SAMV71Q21RT (Cortex-M7, 300 MHz, int16) and 4.4–6.0 ms on a Vorago VA41630 (Cortex-M4, 100 MHz, int16). These are *estimates*; the four COTS boards above are measured directly.

Table 7
BASELINE COMPARISON (SENSOR-BALANCED TRAIN; FULL 54,371-TILE VAL).

| model | G-mean | sens | spec | prec |
|---|---|---|---|---|
| intensity threshold | 0.684 | 0.833 | 0.561 | 0.666 |
| logistic (raw) | 0.685 | 0.761 | 0.617 | 0.676 |
| logistic (PCA) | 0.702 | 0.762 | 0.647 | 0.693 |
| kNN $k$=9 (PCA) | 0.698 | 0.549 | 0.888 | 0.837 |
| SVM-RBF (PCA) | 0.858 | 0.888 | 0.829 | 0.845 |
| random forest (PCA) | 0.870 | 0.932 | 0.812 | 0.839 |
| MLP (PCA) | 0.870 | 0.921 | 0.822 | 0.844 |
| XGBoost (PCA) | **0.884** | 0.912 | 0.857 | 0.870 |
| SRPR poly (ours) | 0.826 | 0.785 | 0.870 | 0.864 |

Table 8
DEPLOYABILITY OF THE HEAD AT MATCHED (SHARED) PCA FRONT END.

| head | params | int8 KB | G-mean | deployable |
|---|---|---|---|---|
| SRPR poly deg-2 | 2,448 | 2.4 | 0.826 | yes (integer) |
| int8 MLP 48-64-32 | 5,282 | 5.3 | **0.870** | yes (integer) |
| random forest | $10^5$–$10^6$ | $\gg$100 | 0.870 | no |
| XGBoost | $10^5$–$10^6$ | $\gg$100 | **0.884** | no |

Table 10
RETIRED INSTRUCTIONS PER INFERENCE AND FLIGHT-HARDWARE PROJECTION.

| build | instr/inference | time |
|---|---|---|
| M4F full f32 | 784,969 | 5.1–7.0 ms @168 MHz |
| M4F full int16 | 401,063 | 2.6–3.6 ms @168 MHz |
| M0 small f32 | 5,619,282 | soft-float bound |
| M0 small int16 | 469,593 | 9.8–17.6 ms @48 MHz |
| SAMV71Q21RT M7 int16 | – | **1.3–1.9 ms** @300 MHz |
| VA41630 M4 int16 | – | 4.4–6.0 ms @100 MHz |

## VIII. ROBUSTNESS

*a) Coarse extent.:* The detector classifies tiles, not pixels. Applied as a dense $40 \times 40$ sliding window over each held-out $256 \times 256$ scene and upsampled, its decisions form a coarse mask: macro pixel IoU 0.38 (Dice 0.50), recall 0.81, precision 0.39 over 1,615 scenes (Table 11). The high recall / low precision reflects the coarse resolution and the $\geq 5\%$ tile-labelling rule, so the output is a *screening* map, not segmentation. Fig. 5 shows one success and one false-alarm failure.

*b) Radiometric robustness.:* Perturbing the held-out tiles with a global gain and offset (Table 11), the G-mean stays within 0.02 of nominal for gains in $\pm 1.5$ dB and offsets in $\pm 0.08$, and degrades markedly beyond ($-3$ dB: 0.72; $+6$ dB: 0.71; $\pm 0.15$ offset: 0.75–0.80). On-orbit radiometric calibration and incidence-angle normalisation must therefore be accurate to roughly $\pm 1.5$ dB; the observed PALSAR/Sentinel gap (Table 3) is the same effect under a real sensor change.

## IX. DETERMINISM, TRANSPARENCY AND AUDITABILITY

Two properties separate the polynomial detector from the higher-accuracy baselines and motivate it for flight software. *Deterministic training:* the head is fit by a convex stage-1 ridge (closed form) followed by deterministic coordinate descent with no random initialisation, so re-training reproduces the

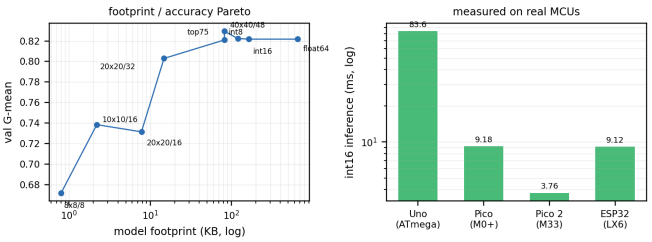



Figure 4. (a) Validation G-mean versus model footprint for the quantization and resolution-ladder variants. (b) Measured int16 inference latency per tile on the four microcontrollers (log scale).

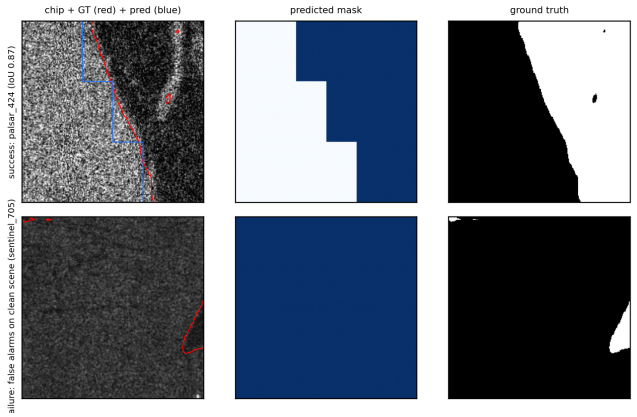



Figure 5. Qualitative dense-window output. Top: a detected slick (predicted contour, blue) aligned with the ground truth (red). Bottom: a false-alarm failure on a clean scene, where the predicted mask floods the tile.

model; forests, boosting and MLPs optimise a stochastic, non-convex objective. *Deterministic, input-independent inference:* the decision is a fixed arithmetic circuit ($48 \times 1600$ PCA multiply–accumulates, 1224 monomials, 2448 head multiply–accumulates), independent of the input. In instruction-accurate simulation the integer path retires the same 401,063 instructions for two very different inputs, so the work — and the worst-case execution time — does not depend on the sample, unlike tree ensembles whose path length is input-dependent; the int32 projection is exact. *Auditability (not physical-law discovery):* the model is an explicit degree-2 polynomial with one stored coefficient per term (Fig. 3; linear PC2 $-0.96$, PC2$^2$ $+0.56$, PC3$^2$ $+0.46$ are the largest weights). A prediction can be decomposed and checked term by term; because the basis is collinear and the inputs are transformed, term-level readings are *illustrative, not identifiable*. The engineering value is auditability, not the recovery of a physical law — a property unavailable in an RF/XGBoost ensemble or an MLP.

## X. Discussion

**What the study establishes.** On a public spaceborne-SAR benchmark, a frozen-PCA + polynomial tile detector compresses to 82 KB (int8; or 15 KB at reduced resolution) and executes in single-digit milliseconds on real 32-bit microcontrollers and 1.3–1.9 ms (simulation) on a rad-tolerant Cortex-M7. The compute and memory budget for on-board oil-spill screening is thus not a blocker.

Table 11
Robustness. (a) Coarse extent mapping vs. SOS pixel masks (dense $40 \times 40$ window); Dice is 0.50/0.47/0.53 for all/PALSAR/Sentinel-1. (b) Radiometric sensitivity (full val; nominal G-mean 0.822).

| **(a) Coarse extent mapping** scenes | IoU | precision | recall |
|---|---|---|---|
| all 1,615 | 0.382 | 0.393 | 0.812 |
| PALSAR (776) | 0.340 | 0.337 | 0.794 |
| Sentinel-1 (839) | 0.420 | 0.444 | 0.828 |
| **(b) Radiometric sensitivity** gain (dB) | G-mean | offset | G-mean |
| $-6$ | 0.454 | $-0.15$ | 0.747 |
| $-3$ | 0.720 | $-0.08$ | 0.787 |
| $-1.5$ | 0.783 | 0 | 0.822 |
| 0 | 0.822 | $+0.08$ | 0.827 |
| $+1.5$ | 0.829 | $+0.15$ | 0.796 |
| $+3$ | 0.807 | | |
| $+6$ | 0.711 | | |

**Threats to validity.** (i) *Task:* tile presence, not segmentation (coarse-mask IoU 0.38). (ii) *Data:* curated benchmark, negatives from oil-scene background; the stock split was PALSAR-only (fixed by sensor-balanced training). (iii) *External:* radiometric tolerance $\sim \pm 1.5$ dB gain / $\pm 0.08$ offset, and the PALSAR/Sentinel gap is the same effect; on-orbit calibration would be required. (iv) *Operational:* look-alike false alarms (biogenic films, rain, low-wind) are the crux and remain unvalidated. (v) *Accuracy:* the polynomial head is not the accuracy leader; an int8 MLP is more accurate at a similar footprint, so the case for the polynomial rests on determinism and auditability.

**Path to deployment.** (1) Independent validation with real look-alike labels and a held-out region/event; (2) a radiometric calibration and incidence-angle normalisation pipeline from raw SAR to the classifier input; (3) a radiation-tolerant evaluation board with real-silicon timing/energy and SEU mitigation; (4) integration with the SAR image chain and a downlink budget; and (5) a CubeSat in-orbit demonstration benchmarked against the ground pipeline.

## XI. Conclusion

We presented a feasibility study for on-board SAR oil-spill screening with an explicit, deterministic polynomial detector. It reaches G-mean 0.822 on the Deep-SAR SOS benchmark, compresses to 82 KB (int8) — 15 KB at reduced resolution for an ATmega328P — and runs in 3.8–9.2 ms on real 32-bit microcontrollers and 1.3–1.9 ms on a simulated rad-tolerant Cortex-M7. Ablations confirm the rank, degree, rescaling and quantization choices; a real-board comparison plus iso-footprint baselines delimit the deployment envelope; and the same battery of checks maps the boundary of the claim: coarse tile-level extent, bounded radiometric tolerance, unvalidated look-alike false alarms, and a baseline in which lightweight deployability is traded against a few G-mean points. The polynomial head is the only candidate that is simultaneously microcontroller-deployable, deterministic and auditable term-by-term. We do not claim an operational detector, but a quan-

tified feasibility envelope, an honest baseline, and a concrete path to a flight demonstration.

## ACKNOWLEDGMENT

The author thanks the providers of the refined Deep-SAR SOS dataset and the open-source TinyML toolchains used for the on-device measurements.

## DECLARATION OF AI AND AI-ASSISTED TECHNOLOGIES IN THE WRITING PROCESS

During the preparation of this work the author(s) used OpenCode (DeepSeek-Flash), an AI assistant, to create the initial draft of the manuscript. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication. All figures were generated from model and system output using Python code and were not created or altered by generative AI.

## REFERENCES

[1] Q. Zhu *et al.*, "Oil Spill Contextual and Boundary-Supervised Detection Network Based on Marine SAR Images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–10, 2022. https://doi.org/10.1109/TGRS.2021.3115492

[2] D. Zuenko and I. Khaidarova, "Refined Deep-SAR Oil Spill (SOS) dataset," Zenodo, 2025. https://doi.org/10.5281/zenodo.15298010

[3] European Maritime Safety Agency, "CleanSeaNet — Satellite-based oil spill and vessel detection service." [Online]. Available: https://www.emsa.europa.eu/csn-menu.html

[4] C. Wu *et al.*, "A Comprehensive Survey on Orbital Edge Computing: Systems, Applications, and Algorithms," *Chinese J. Aeronautics*, 2024. https://doi.org/10.1016/j.cja.2024.11.026

[5] T. Pfandzelter and D. Bermbach, "Can Orbital Servers Provide Mars-Wide Edge Computing?," in *Proc. ACM MobiCom Workshop SatCom*, 2023. https://doi.org/10.1145/3614454.3622997

[6] Q. Zhang *et al.*, "Resource-efficient In-orbit Detection of Earth Objects," in *Proc. IEEE INFOCOM*, 2024. https://doi.org/10.1109/INFOCOM52122.2024.10621328

[7] A. Furutanpey *et al.*, "FOOL: Addressing the Downlink Bottleneck in Satellite Computing with Neural Feature Compression," *IEEE Trans. Mobile Comput.*, 2025. https://doi.org/10.1109/TMC.2025.3544516

[8] M. Plumridge *et al.*, "Rapid Distributed Fine-tuning of a Segmentation Model Onboard Satellites," in *Proc. IEEE IPAS*, 2025. https://doi.org/10.1109/IPAS63548.2025.10924497

[9] G. Inzerillo, D. Valsesia and E. Magli, "Efficient Onboard Multi-Task AI Architecture Based on Self-Supervised Learning," *IEEE JSTARS*, 2024. https://doi.org/10.1109/JSTARS.2024.3502776

[10] S. Lu *et al.*, "WISE: A Lightweight, Weakly-Supervised Model for Onboard Fire Smoke Detection and Localization," arXiv:2609.18227, 2026. https://arxiv.org/abs/2609.18227

[11] F. M. Bianchi, M. M. Espeseth and N. Borch, "Large-Scale Detection and Categorization of Oil Spills from SAR Images with Deep Learning," *Remote Sensing*, vol. 12, no. 14, p. 2260, 2020. https://doi.org/10.3390/rs12142260

[12] E. Amri *et al.*, "Deep Learning Based Automatic Detection of Offshore Oil Slicks Using SAR Data and Contextual Information," in *Proc. SPIE Remote Sensing*, 2021. https://doi.org/10.1117/12.2598032

[13] A. Ramanathapura Satyanarayana and M. A. Dhali, "Oil Spill Segmentation Using Deep Encoder-Decoder Models," in *Proc. ICPRAM*, 2025. https://doi.org/10.5220/0013259600003905

[14] C. Banbury *et al.*, "MicroNets: Neural Network Architectures for Deploying TinyML Applications on Commodity Microcontrollers," in *Proc. MLSys*, 2021. https://arxiv.org/abs/2010.11267

[15] A. Gholami, S. Kim, Z. Dong, Z. Yao, M. W. Mahoney and K. Keutzer, "A Survey of Quantization Methods for Efficient Neural Network Inference," in *Low-Power Computer Vision*, Chapman and Hall/CRC, 2022, pp. 291–326. https://doi.org/10.1201/9781003162810-13

[16] P.-E. Novac *et al.*, "Quantization and Deployment of Deep Neural Networks on Microcontrollers," *Sensors*, vol. 21, no. 9, p. 2984, 2021. https://doi.org/10.3390/s21092984

[17] V. A. Martirosyan, "Efficient and Interpretable On-Device Polynomial Inference for Multi-Variable Ocean Profiling," in *Proc. eICocean*, 2027, in press.