QML · pulsar anomaly detection

Asadbek Ismoilov

1 · PROBLEM

what and why

Radio pulsar surveys generate candidate images faster than they can be triaged, and the positive class (pulsars) is rare — about 2% of samples. Classical CNNs already handle this well at scale, but offer no path to the parameter- and data-efficiency gains needed for edge deployment at telescope sites or for low-SNR regimes where labeled anomalies are scarce. Quantum machine learning offers a genuine efficiency argument here: HTRU-1's 32×32×3 images are small enough to fit current NISQ qubit budgets via amplitude encoding, so the question is whether a quantum pipeline can match classical accuracy at a fraction of the parameters and data.

2 · APPROACH

model / method, and why this one

Quanvolutional Neural Networks (QCNNs) are primary: local sliding-window quantum circuits replace classical convolution filters as feature extractors ahead of standard neural layers. Comparators: Variational Quantum Classifiers (VQCs) and Quantum Support Vector Machines (QSVMs). QCNN is chosen because it has demonstrated barren-plateau mitigation (localized cost functions + tensor-network initialization) at the accuracy/parameter tradeoff needed — 98.7% accuracy at 45 parameters versus ~120,000 for a comparable classical baseline.

3 · DATASET

source, size, preprocessing

HTRU-1 (target): 60,000 samples, 32×32×3 pulsar-survey images; 40k/10k/10k split; ~2% positive class. Preprocessing: min-max scaling, random flips, rotation ±15°. HTRU-2 (feature-quality benchmark): 17,898 samples, 8 tabular features, ~9% positive. MNIST (0/1) and CIFAR-10 (airplane/automobile) for calibration only — go/no-go gates before HTRU-1 runs.

4 · METRICS

how you'll know it worked

how you'll know it worked Primary: classification accuracy and anomaly-detection recall/precision on HTRU-1 vs HTRU-2 ceiling and classical CNN/ResNet-lite. Secondary: gradient variance scaling with qubit count, parameter count vs accuracy, and accuracy retention under 1% depolarizing noise (target >90%).

5 · COMPUTE

training time, memory, dependencies

Amplitude encoding: n = ■log■ N■ qubits. Circuit depth bounded by hierarchical pooling. Training: batch 32, 150 epochs, TNI pre-training. Measured runtimes ~2.3–18.3 hrs (CPU) / ~10.1–33.7 hrs (GPU) across feature configurations. GPU-accelerated simulation (e.g. NVIDIA CUDA-Q) treated as required.

6 · DELIVERABLES

model, paper, demo

- Trained QCNN for HTRU-1 plus VQC/QSVM comparators.

- · AdaInit initialization protocol adapted to this task.
- · Write-up covering accuracy, parameter efficiency, and noise resilience.
- · Notebook reproducing MNIST/CIFAR-10 calibration and HTRU-1 end-to-end.

7 · RISK

what breaks first, and plan B

what breaks first, and plan B Trainability: barren plateaus and random-init traps. Mitigated via localized costs, TNI warm-start, and AdaInit. Hardware: decoherence and gate error — bound depth to O(log n). Data: ~2% positive class risks overfitting; HTRU-2 used as a feature-quality ceiling to detect when the image pipeline — not the quantum circuit — is the bottleneck.