Yang Hao 杨豪

Model Quantization & Deployment Lead • ML Infrastructure • Autonomous Driving & Edge AI

Email howellyang@buaa.edu.cn
Web howellyang.com
GitHub github.com/Howell-Yang

7+ years taking deep-learning models from training to production automotive chips. At QCraft I own quantization and on-device deployment for about 19 production driving and parking models (BEV perception, occupancy, vector maps, temporal tracking, prediction-planning, end-to-end) on Horizon J5/J6M/J6P, NVIDIA Orin, Qualcomm and AXERA. Depth from LSB-level fixed-point error and quantization-aware distillation to on-board latency modeling, built into toolchains, release gates and team process. I lead a 6-engineer team and am technical lead of a new-chip program.

≈19	2 → 0.5	⩽ 2 LSB	0.7%	4,273	2 wk → 3–5 d
production models quantized and deployed driving + parking	points of quantization loss PTQ → QAT five perception models	worst-case LUT operator error 600 modules 85% fewer tables	on-board latency prediction error 19.38 ms predicted 19.25 measured	production model conversions build time 30–60 → 5 min	per-model quantization adaptation layered toolchain + AI auto-adaptation

2012 – 2019	2019.04	2021.07	2022.10	2024	2025	2026
Beihang BS + MS Automation Pattern Recognition	**Yitu Technology** face-recognition engine then chip quantization	**Tencent Maps** TensorRT INT8 on-device SDK	**Joined QCraft** QNN Web production releases	**Promoted to L5** J6M toolchain AutoQuant	**Quantization lead** team lead distillation QAT	**Direct reports new-chip tech lead** large-scale QAD platform framework

EXPERIENCE AND REPRESENTATIVE PROJECTS

QCraft 轻舟智航 Model Quantization & Deployment Lead, Perception ML-Infra

Oct 2022 – present

Joined as Machine Learning Infrastructure Engineer; promoted to L5 in 2024; quantization lead and team lead since 2025 (direct reports from 2026).

WHERE THE PROJECTS SIT IN THE DEPLOYMENT PIPELINE

Float training 5	Quantization 1 2	Export IR 3	Multi-chip build 4	Alignment and gates 3	Board and vehicle 5 6
training speed-up shared backbone	PTQ / QAT / QAD case-level float parity	fixed-point LUTs operator parity	QNN Web four chip families	stage-by-stage checks evidence levels	latency prediction new-chip bring-up

1 Float-consistent quantization: four generations of method

2023 – 2026

PROBLEM Production tolerates almost no regression (an AEB target detected one frame late is a failure), yet QAT trained on the task loss alone shifts the precision/recall operating point, so what ships is effectively a retrained model.

- **Scale-search PTQ** (2023, patent filed): searches quantization parameters in stages against cosine similarity to the float outputs: calibration method × parameter grid → layer-wise backward/forward search → layer-wise quantization. On a traffic-light detector, default calibration gave mAP 0.747 and layer-wise quantization restored it to **0.803 (float 0.800)**.
- **LSQ-based AutoQuant** (2024) for CNN + Transformer hybrids (vector map, temporal tracking): learns step sizes, constrains weight drift with AdaRound-style rounding, and supervises with float intermediate features, with cross-layer equalization (CLE) and LayerNorm–Linear fusion applied first.
- **Distillation QAT → large-scale QAD** (2025–2026): float model as teacher, PTQ model as the start, task loss plus feature/output distillation. On the end-to-end planner, single-variable experiments (25 key metrics, 388 shared scalars) showed that scene-boundary state reset, an independent teacher history and late-stage weight anchoring are all required.

RESULT Quantization loss on five production perception models fell from about 2 points to about **0.5**; the quantized end-to-end planner went to on-vehicle testing; model owners have started running auto-QAT releases themselves.

2 Quantization error accumulation in a prediction-planning model (QuestNet v3)

2025 – 2026

PROBLEM After quantization the model produced abnormal trajectories; PTQ and standard QAT both failed and a production delivery was blocked.

- Per-layer and cumulative error analysis showed the error **compounding through the deep layers**; sensitive and robust layers are treated differently. Experiments ruled out “freeze some layers and fine-tune” in favor of **two-stage quantization with full QAT**.
- Drawing on the chip's operator characteristics, moved the whole pipeline to **INT16** for numeric headroom against the accumulation.

RESULT Trajectory anomalies eliminated; merged to mainline and release-validated. “Mid-fusion models need full QAT” is now team practice.

3 Fixed-point numerical correctness: adaptive LUTs and layered alignment

2024 – 2026

- **Adaptive lookup tables**: on the NPU, rsqrt, reciprocal, exp and sigmoid run as fixed-point piecewise LUTs. The algorithm measures error on the chip's actual fixed-point output codes (not a float approximation), keeps the original 6-table boundaries, recursively bisects the worst segment, and blocks export if the budget cannot meet the bound. Across 600 LUT modules in four production graphs: **0.217 LSB** mean, **2 LSB** max (uniform 126-table: 11,381 LSB max), with **85%** fewer tables.
- **Layered alignment**: Float → QAT → in-house IR → LibTorch → HBIR → HBM → on-board post-processing, each stage compared only with the previous one on identical inputs, so a mismatch has exactly one possible source. Pinned down an rsqrt LUT split point in a norm layer, an upsample align_corners mismatch and pose-precision bugs.
- **Release gates**: results graded by evidence level (source / build / simulation / board / vehicle), default-off feature flags, and a regression set of past quantization failures re-run before every release; caught issues pre-release on three automaker programs.

4 Toolchain and platform: from one-click conversion to a quantization-deployment platform

2022 – 2026

- **QNN Web**, a one-click multi-chip conversion service (gRPC front/back end, configs generated from the ONNX graph, TensorRT timing cache; later a job database, queueing and MCP endpoints for AI agents): **4,273** production conversions, build time from 30–60 min to about 5, about 80% less manual effort.
- Split multi-chip quantization into a **generic layer, an AI auto-adaptation layer and a platform layer**; per-model adaptation went from about 2 weeks to **3–5 days**. Led the team onto Qualcomm (model splitting and latency tuning; inference latency cut by more than 50%), TensorRT QDQ INT8 and VLM quantization on Horizon J6P.
- Since Nov 2026, leading the target architecture of a **platform-style quantization-deployment framework** (design draft, roadmap to Q2 2027): fixed-point simulation as the single source of truth, seven versioned contracts (artifact identity, IR metadata, IO contract, golden IO, precision profile, target capability, thresholds) across the training, conversion and on-vehicle codebases, blocking gates at every hop, and a sensitivity × latency-cost loop that plans mixed precision.

5 Latency modeling, model architecture and training efficiency

2025 – 2026

- **Structural latency model**: predicts on-board latency from module structure; the occupancy model was predicted at 19.38 ms and measured at 19.25 ms (**0.7%** error). It ruled out shipping "wider range + 10 Hz" before any code was written and confirmed that MapFormer query merging saves 2.1–3.3 ms.
- **OneModel phase 2** (in progress): one shared image backbone and neck for five perception tasks and 9 cameras, estimated to save 5–9 ms per frame. Backbone sized by PCA redundancy analysis of the five tasks' checkpoint features; pretraining weights synthesized by channel alignment plus a closed-form linear solve; quantization-friendly ReLU6 with surrogate gradients; receptive-field center offset fixed.
- Migrated BEV occupancy training to PyTorch 2.x: 1.56 s → 0.71 s per iteration (**−54%**; V100, batch 16, AMP), 9% less GPU memory, scaling benchmarked from 1 to 112 GPUs.
- **End-to-end driving model architecture** (design proposal, not yet trained): a camera-first, token-based design: 9 cameras tokenized through the shared encoder, a streaming transformer with pose-aware memory, a multi-mode plan head parameterized by curvature and acceleration knots, with a distill-then-INT8 path designed in from the start.

6 Team leadership and program delivery

2025 – 2026

- Lead a **6-engineer** quantization and deployment team (dotted-line from 2025, direct reports from 2026): an owner and a backup for each of about 19 production models, quarterly goals with P0/P1 priorities, work assigned by urgency × difficulty. In Q2 2026, with four production programs in parallel on a two-week release cadence, no release slipped because of quantization.
- **Technical lead** since June 2026 for the AXERA chip program (validated on the M97 700-TOPS dev board; target chip M95, 350 TOPS): staged bring-up plan, model adaptation and quantization, road-test issue triage. The urban NOA v4 stack runs on test vehicles, with no frame drops on key models.
- **AI-assisted automation**: a Feishu bot plus coding-agent skills adapt new model features for quantization (60 person-day project). Maintain a 33-page knowledge base, run monthly quantization sessions and coach engineers to independent model ownership.

EARLIER EXPERIENCE

Tencent Maps 腾讯地图 Algorithm Engineer, CSIG Map Platform

Jul 2021 – Oct 2022

- **TensorRT INT8 deployment**: built the quantization flow and conversion tooling for cloud GPU models (bias correction, weight equalization, layer-wise calibration); cosine similarity to float above 0.999, 20% lower latency, 10% lower GPU utilization and memory, about RMB 3M of compute saved. The calibration tool is open-sourced as onnx2trt.
- **On-device perception SDK**: multi-pipeline AR-navigation framework with automatic profiling; CPU utilization down to 46% and 80 ms end-to-end per frame, shipped on a production vehicle. Privacy masking for map data collection used a 38-MFLOPs face/plate detector (recall 74.0, precision 90.6) with DeepSORT and optical-flow tracking.

Yitu Technology 依图科技 Algorithm Engineer

Apr 2019 – Jul 2021

- Built a C++ face-recognition engine that combines several models, tuning the combination strategy for throughput and recognition rate (used in metro face-pass projects). When the company began building its own AI chip, moved to model quantization: quantization tools, accuracy evaluation and a model zoo.

TECHNICAL SKILLS

Quantization	PTQ, QAT, QAD and distillation, LSQ / AutoQuant, AdaRound, CLE, scale search, mixed INT8/INT16, fixed-point LUT operators, sensitivity analysis, per-layer error attribution
Deployment targets	Horizon J5 / J6M / J6P (Horizon plugin, HBIR/HBM, BPU), NVIDIA Orin (TensorRT, custom plugins, QDQ), Qualcomm (QNN), AXERA M95 / M97
Model types	BEV detection, occupancy, vector maps, sparse temporal detection and tracking, prediction-planning, end-to-end, multi-task perception, VLM
Training and engineering	PyTorch 1.x/2.x, DDP, AMP/BF16, multi-node training (NVIDIA V100/H20, Alibaba PPU); Python, C++, ONNX, TorchScript/LibTorch, gRPC, MCP tool servers, AI coding agents
Languages	Chinese (native), English (technical presentations and documentation)

EDUCATION

Beihang University (BUAA) 2016 – 2019
M.S. in Pattern Recognition

Beihang University (BUAA) 2012 – 2016
B.S. in Automation

PATENT, TALKS AND CUSTOMER ENGAGEMENT

- **Patent filed**: automatic quantization-parameter search framework.
- **Customer engagement**: bilingual quantization-methodology briefing for an overseas automaker; technical workshops with Horizon Robotics and AXERA.
- **Talks**: monthly in-house quantization sessions; company-wide tech-weekly talk.