OBJECT GROUNDING ANALYSIS

CVPR 2026 Tutorial

The IEEE/CVF Conference on Computer Vision and Pattern Recognition 2026

Denver, CO, USA

THE SETUP

Current Jabra Device vs High-End Qualcomm Technology

Fairphone 5
Snapdragon QCM6490. Same SoC family as the Jabra cameras we ship today.

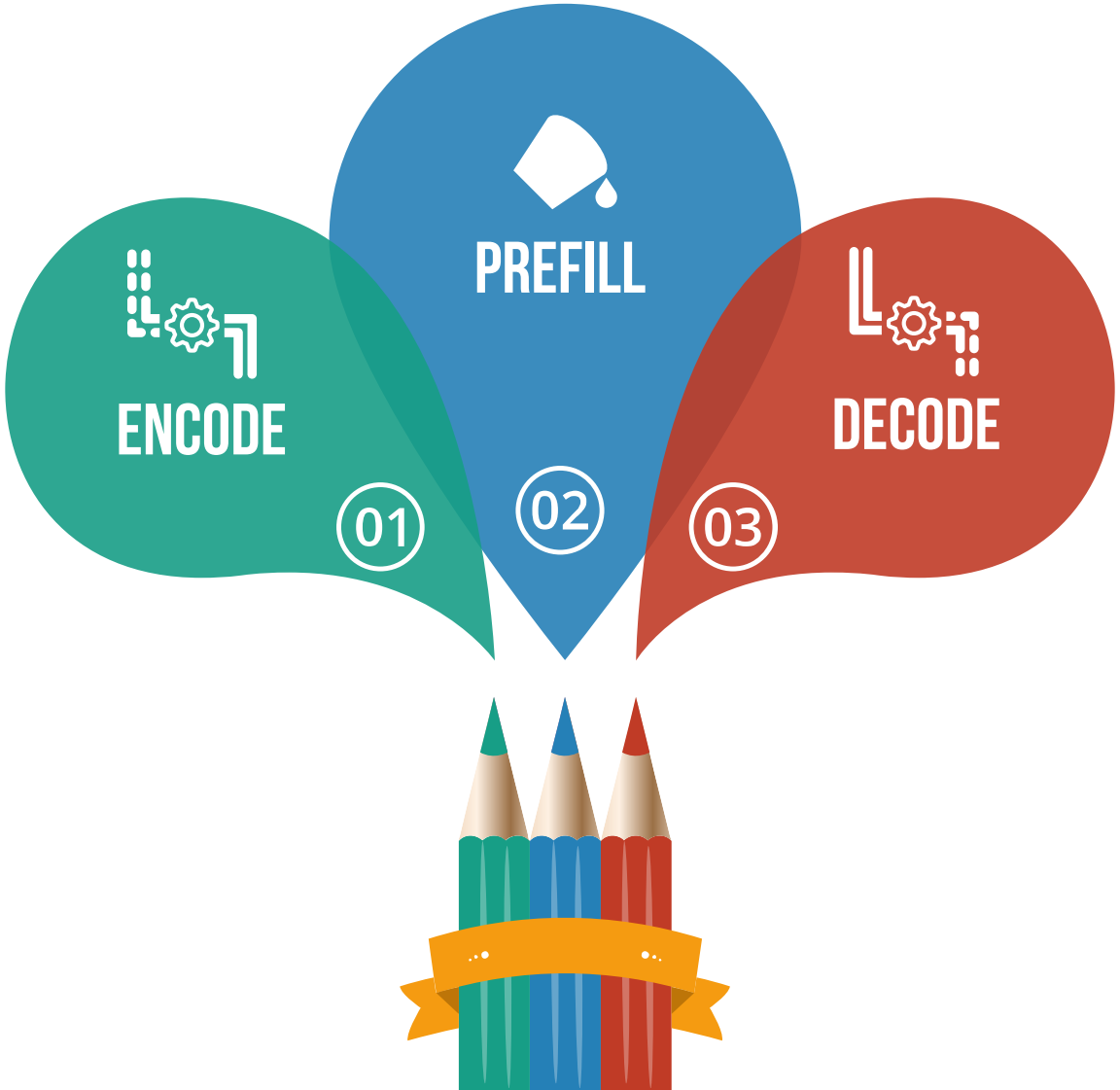
S25 Ultra
Snapdragon 8 Elite (SM8750-AC). The high-end technology provided by Qualcomm.

COCO val2017
Single-person crops. 50 images per config × every knob × both SoCs.

Sweep Total
1,150 runs on FP5 · 2,620 on S25 Ultra · 3,770 total.

WHAT "FAST" MEANS

TTFT = Encode + Prefill



Info 01

Encode

The vision projector runs over patches of the image.

Info 02

Prefill

The LM reads visual + text prompt into KV cache.

Info 03

Decode

The LM streams output tokens — bbox JSON, ~36 tokens.

THE FOUR METRICS

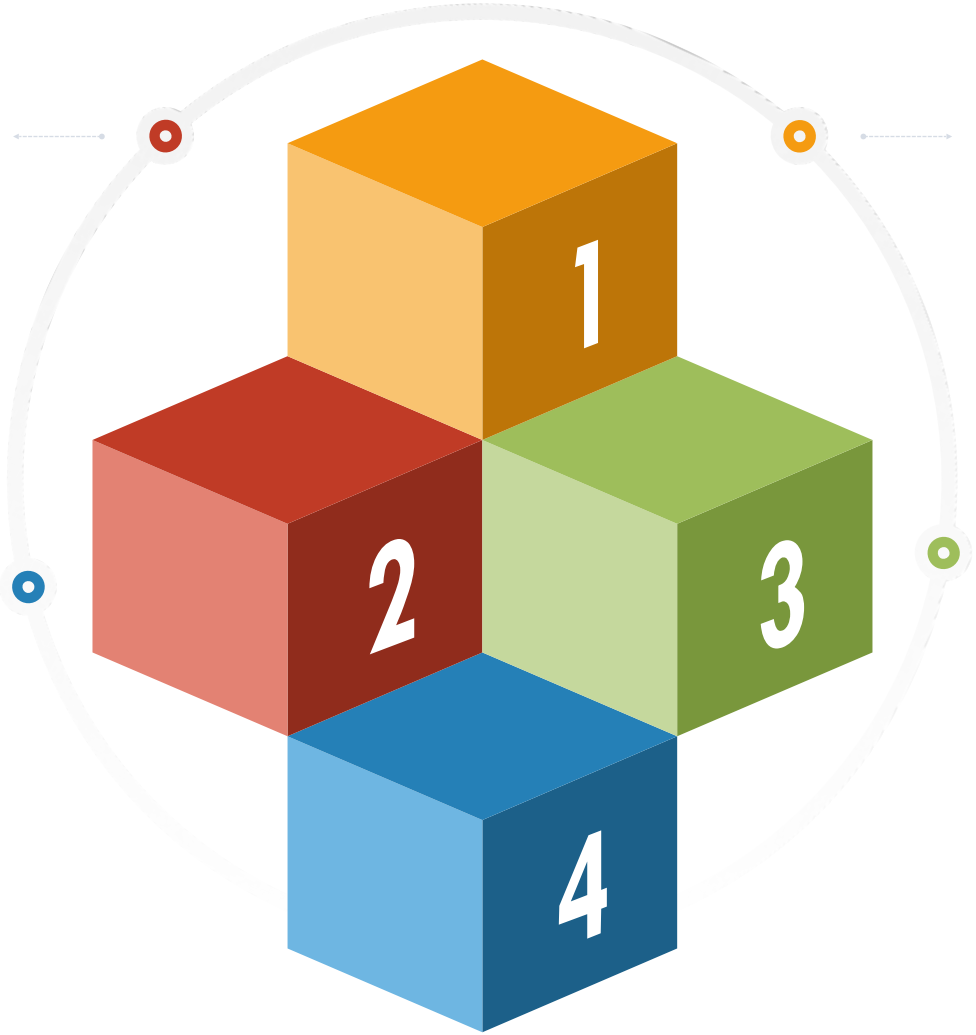
Measurements on Every Run

Total

TTFT + Gen. Wall-clock per image.

TTFT

Encode + prefill. The blank-screen duration the user actually feels.



mAP@.5

Fraction of predicted boxes with IoU ≥ 0.5 vs ground truth. Dataset: COCO val2017 single-person crops.

Gen

Decode time. Short for bbox JSON — about 1.3 s on S25 CPU.

TTFT IS
91%
OF WALL-CLOCK

TTFT 14.0 S · GEN 1.7SEC
CUTTING TOKENS 2×
SAVES ~7SEC

HOW A PHONE "SEES" AN IMAGE

Patch → Merge → Visual Tokens

01 Patches

640 × 480 raw image → split into 16 × 16 px patches.

02 Merge

Each 2 × 2 patch block is merged into one visual token.

03 Tokens

20 × 15 = 300 tokens feed into the LM. Same math for Qwen3.5 and Qwen3-VL.

THE BENCHMARK GRID

Methodology

TOTAL INFERENCE:
3,770 runs across both SoCs

MMPROJ QUANTIZATION:
F16 vs Q8_0

FAMILY:
Qwen3.5 vs Qwen3-VL

VISUAL BUDGET:
3 floors × 9 caps

LM QUANTIZATION:
Q4_0 vs Q8_0

QWEN3-VL 2B BEATS
QWEN3.5 AT EVERY
MATCHED CONFIG

MULTI-LAYER VISION
ROUTING BAKED
INTO QWEN3-VL

Q8 MMPROJ IS EFFICIENT

Same Accuracy and 35% Faster

Inference Time: 21.8 seconds, mAP 0.60

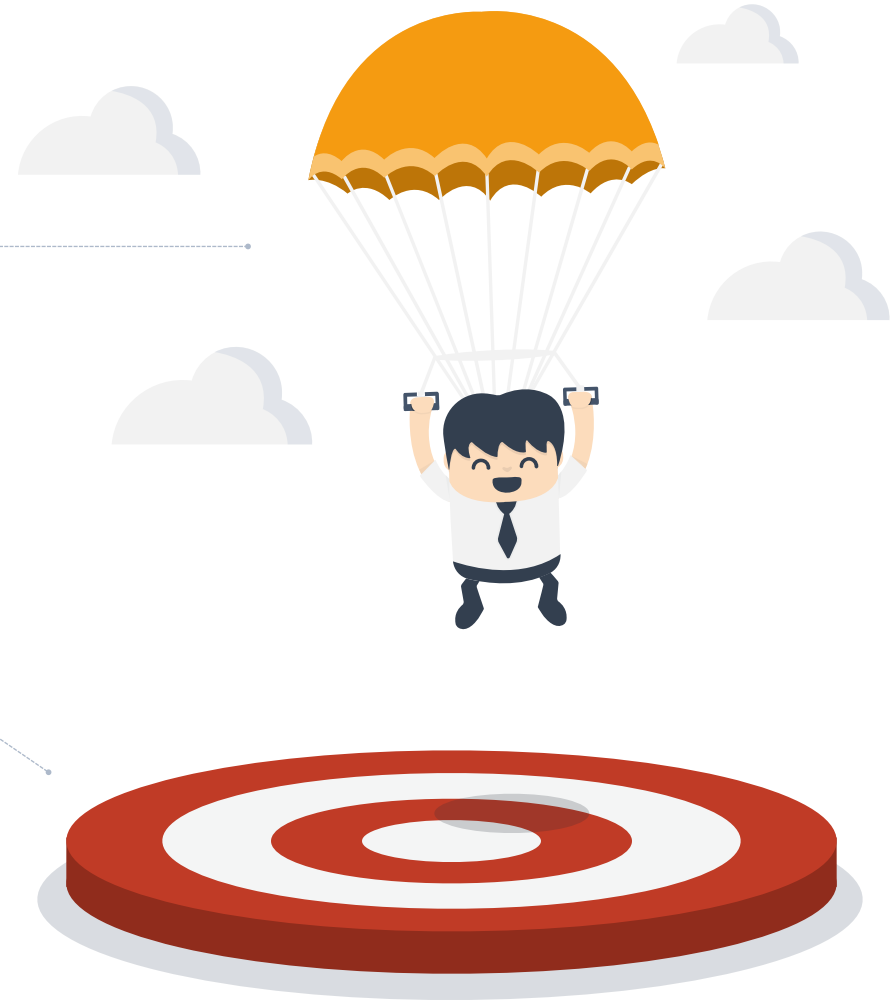
Vision Projector: ~430 MB

F16 MMPROJ

Inference Time: 14.2 seconds, mAP 0.60

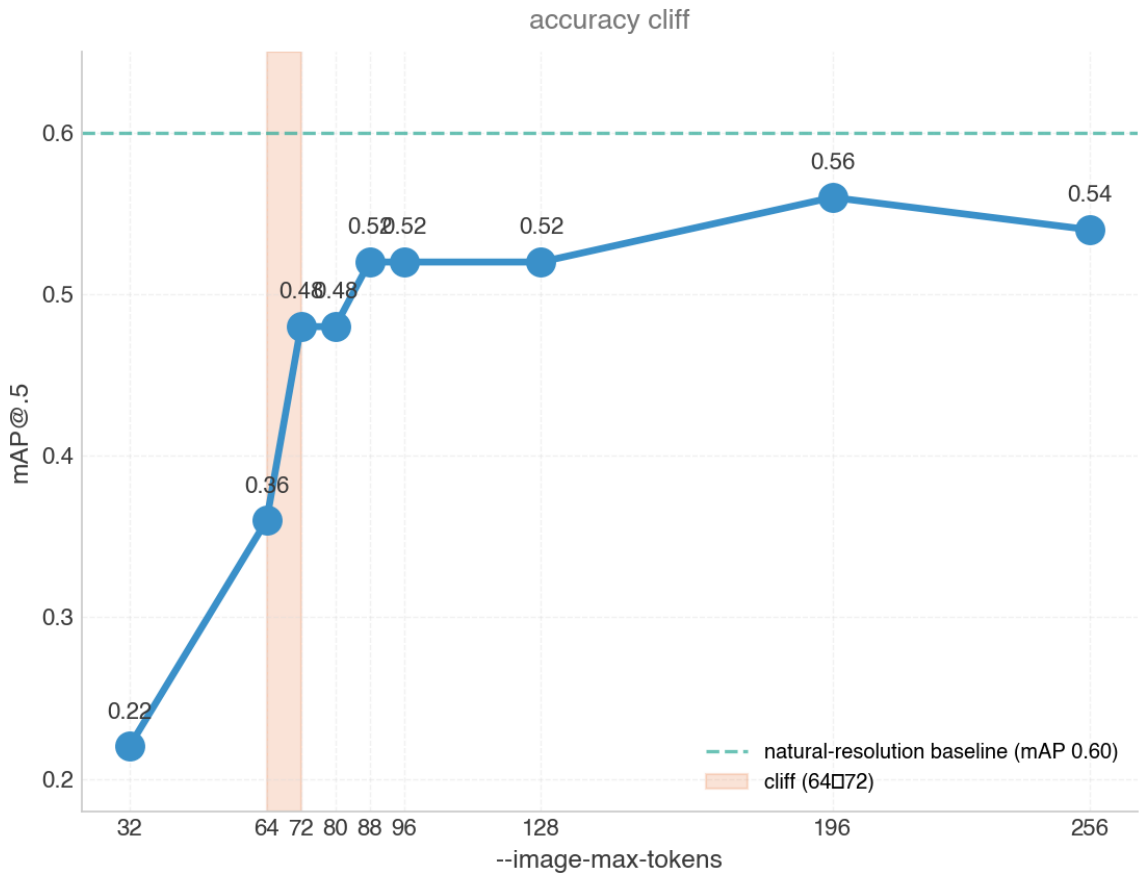
Vision Projector: ~220 MB

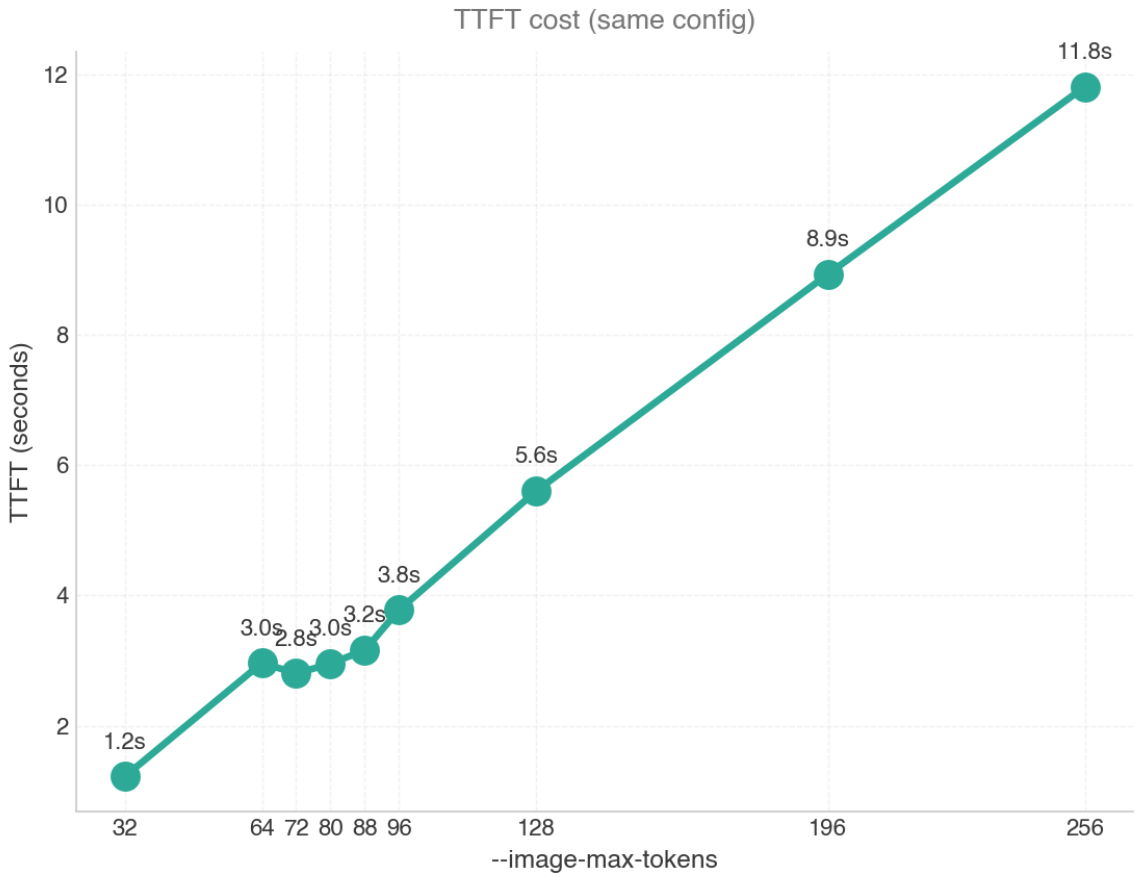
Q8 MMPROJ



THE ACCURACY CLIFF

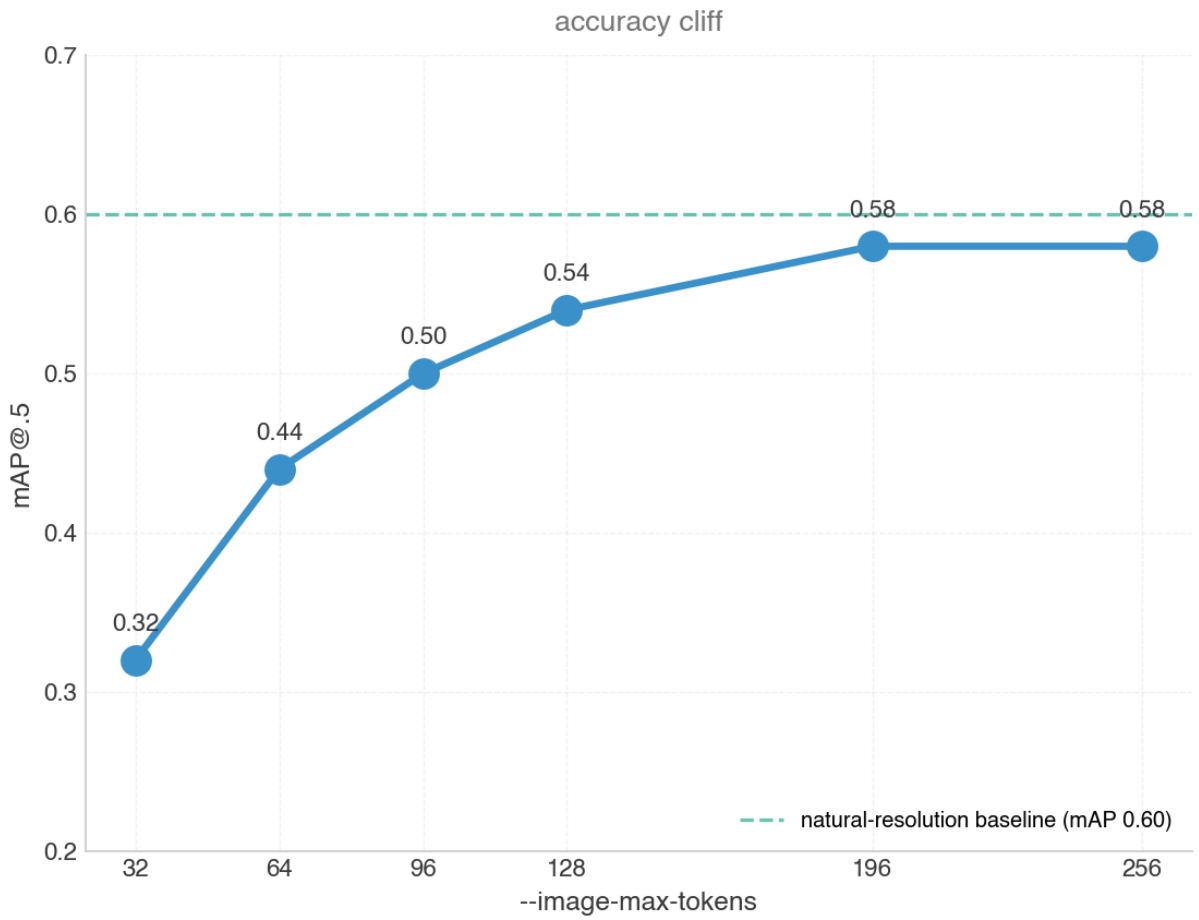
mAP Jumps 0.36 → 0.48 Between 64 and 72 Visual Tokens

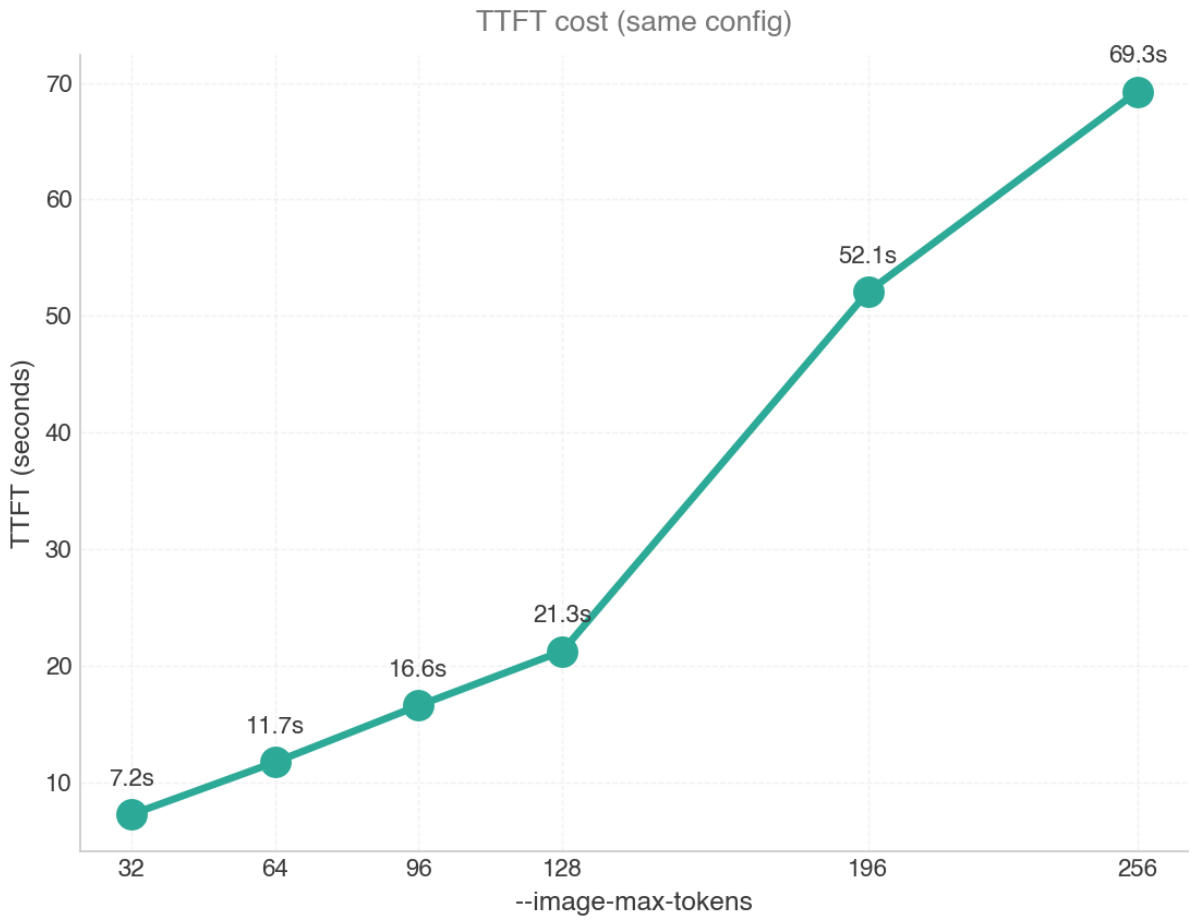




ACCURACY VS TTFT COST

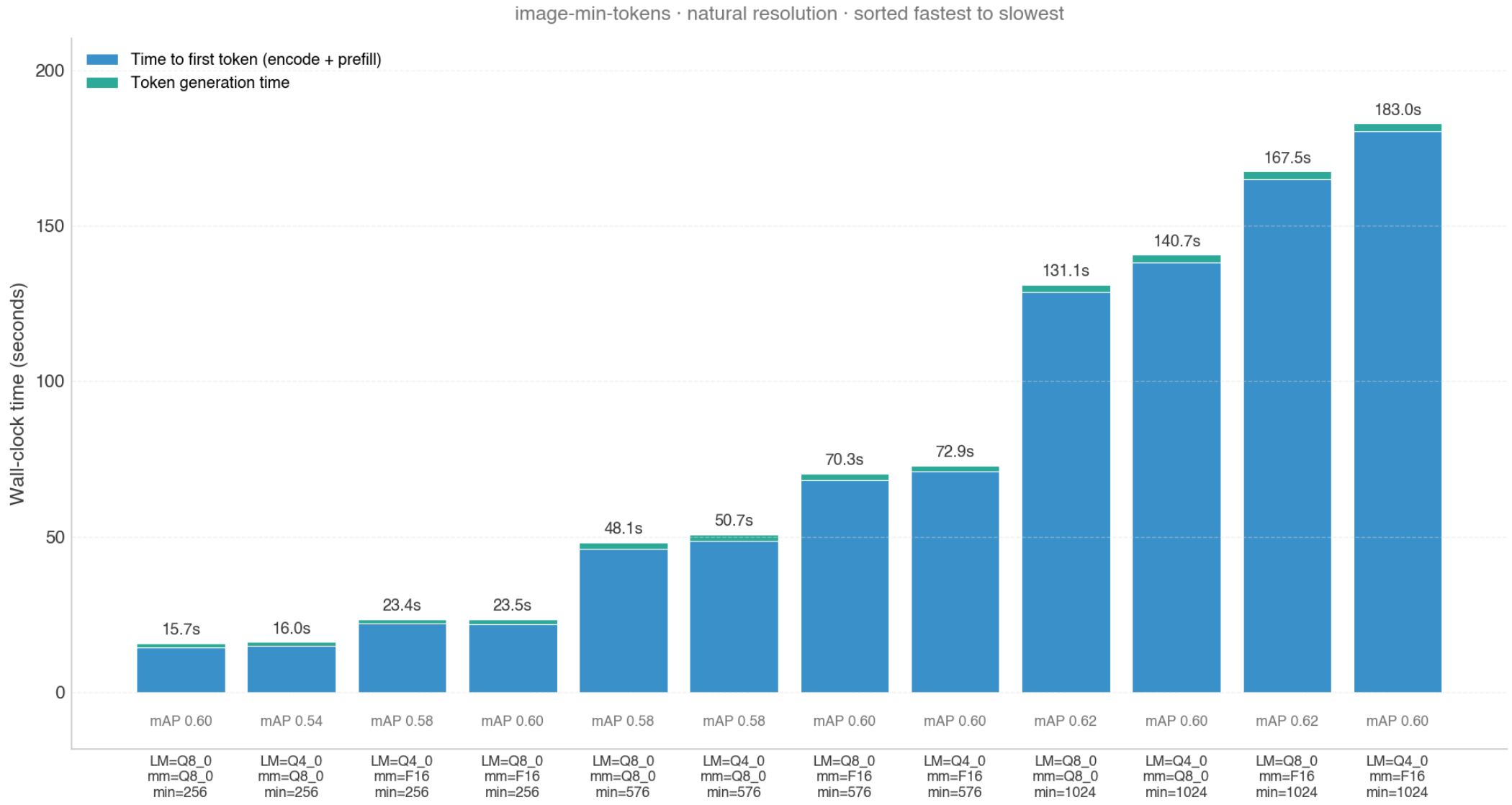
QCM6490 SoC (FP5): 13.5sec @ 72 Tokens vs SM8750-AC SoC (S25): 2.8sec @ 72 Tokens





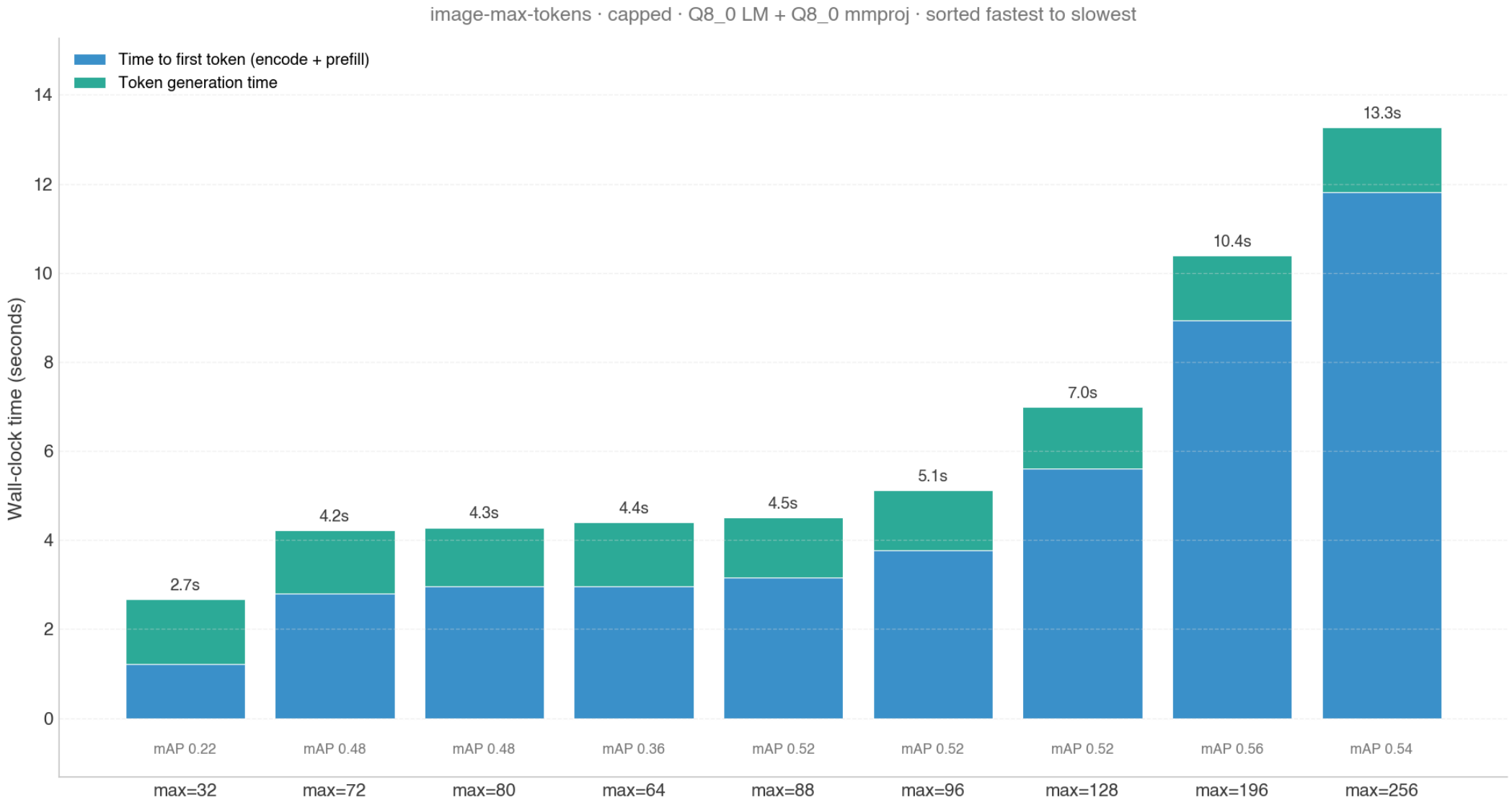
WALL-CLOCK AT NATURAL RESOLUTION

Without a Token Cap: 15.7sec Best, 183sec Worst



WALL-CLOCK WITH THE CAP

Max=72 Gets Us to 4.2 s @ mAP 0.48



OUR RECOMMENDATIONS

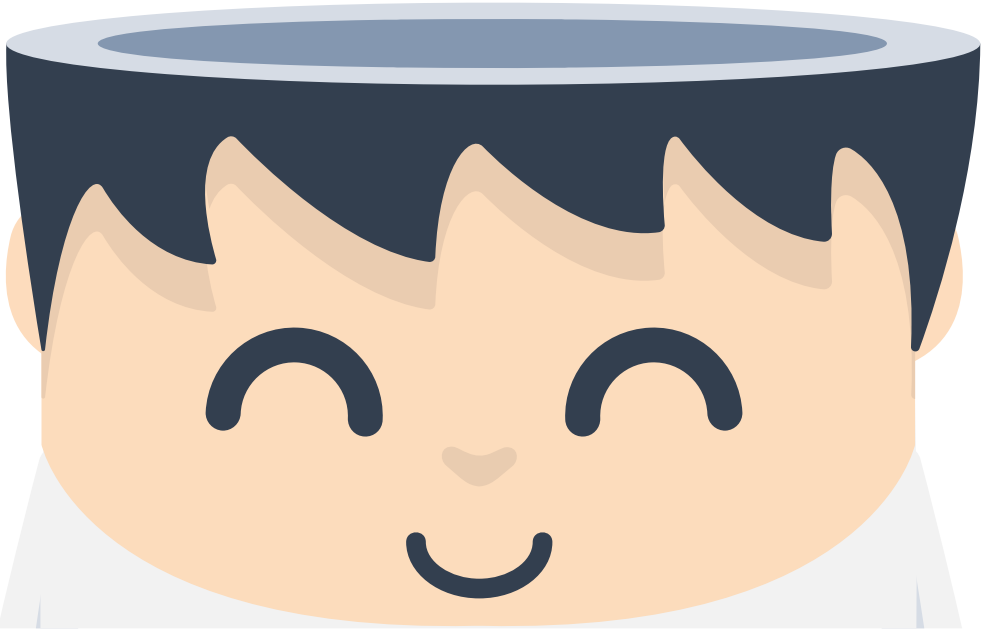
After Benchmark Runs

Model

Qwen3-VL 2B at Q8_0
+ Q8_0 mmproj.

Visual Budget

--image-max-tokens 72.
The cliff edge of the
accuracy curve.



Backend

CPU · -t 4 · --cpu-mask
0x0F (2 Prime + 2 Perf) ·
--no-mmproj-offload ·
-fa · Q8_0 KV cache.

Expected

~3–4 s per capture ·
mAP@.5 ≈ 0.48 on
SM8750-AC.

WHAT DIDN'T WORK

Our Findings

01 GPU Offload

Our quick check: driver kernel-compiler crashes on Adreno 643 (FP5). Works on S25 Adreno 830 (3.4×). Full backend analysis is in the hardware-track deck — CPU still wins.

02 F16 mmproj

No quality gain over Q8 at this image size. Pure cost — 35% more TTFT.

03 GBNF Grammar

Works end-to-end via mtmd but slows small models down. Left disabled.

04 Qwen3.5 0.8B

Faster, but mAP collapses. Hallucinates boxes. Not a win.

05 Hexagon HTP DSP

Our quick check: 8.4 s captures on S25. CPU still wins at 72 tokens.

TAKEAWAYS

Five Things to Remember

TTFT, NOT TOK/S 01 — The user-visible cost is encode + prefill. Optimize that first; LM tok/s improvements barely move the needle.

FAMILY, NOT SIZE 02 — Qwen3-VL 2B beats a faster Qwen3.5 0.8B that hallucinates. Architecture > size.

Q8 MMPROJ 03 — Free win on CPU. Cuts TTFT 35% with zero accuracy cost. No reason to use F16.

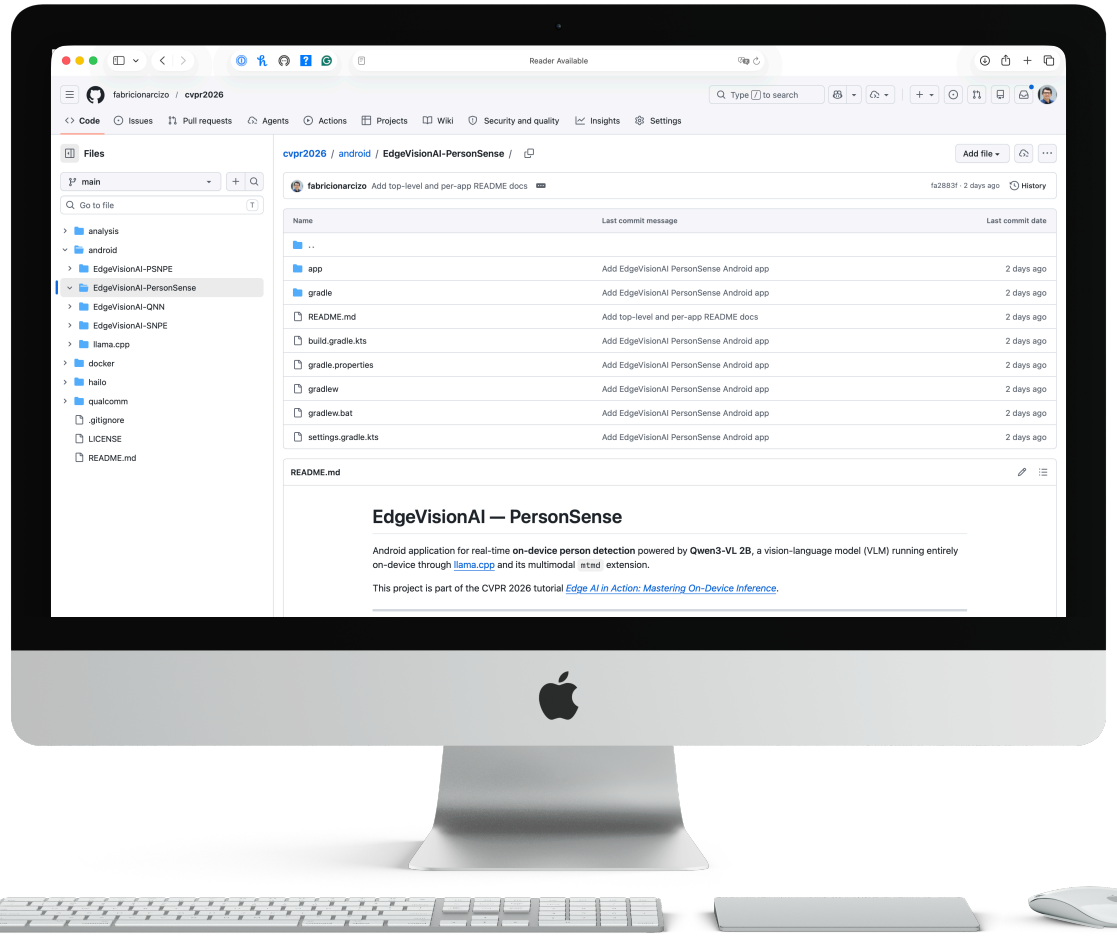
CLIFF AT 72 04 — Below it, accuracy drops. Above it, time increases.

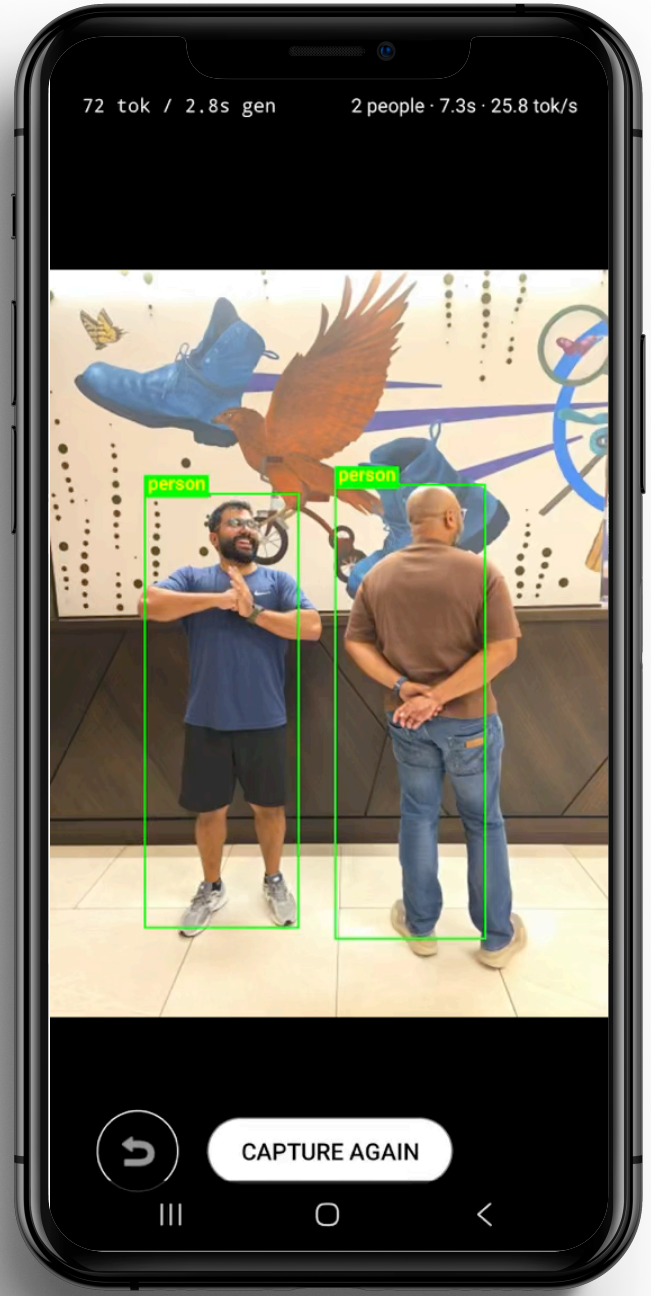
VISUAL TOKEN BUDGET 05 — The dominant parameter.

PERSON SENSE APP

https://github.com/fabricionarcizo/cvpr2026/tree/main/android

SAMSUNG Galaxy S25





QUESTIONS & ANSWERS

THANK YOU!