Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

A Few Points of Rank-1, Several Times the False Accepts: Calibration and the Operating Point of INT8 Face Verification

Muhammad Rizwan Tahir

rizwantahir@rootblocklabs.com

Rootblock Labs

Muhammad Hasnain Amjad

hazen.ai

Research Article

Keywords:

Posted Date: November 25th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11707524/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

A Few Points of Rank-1, Several Times the False Accepts: Calibration and the Operating Point of INT8 Face Verification

Muhammad Rizwan Tahir[1*] and Muhammad Hasnain Amjad[2]

[1*]Rootblock Labs, Dubai, United Arab Emirates.
[2]hazen.ai, Makkah, Saudi Arabia.

*Corresponding author(s). E-mail(s): rizwantahir@rootblocklabs.com;

Abstract

Face recognizers on edge devices are routinely converted to INT8 with post-training quantization (PTQ) and accepted on identification accuracy. We show that this check understates the damage to verification and does not reveal its size. We quantize an ArcFace ResNet-50 with ONNX Runtime's default MinMax calibration from 10 random calibration sets of 512 images. In every draw the false accept rate (FAR) at the threshold tuned on the FP32 model exceeds its $\mathbf{10^{-3}}$ target, by 1.3 to 6.0 times (median 2.2), also when the threshold is fitted on held-out identities. Re-tuning the threshold costs 18–43% of the true accept rate (TAR) at FAR$= \mathbf{10^{-3}}$, while Rank-1 drops only 2.0–3.5 points and the area under the ROC curve at most 0.022. The excess follows the rise and widening of the impostor scores. Percentile calibration is far more stable (FAR 1.1–1.3$\times$ target), ONNX Runtime's default Entropy setting reduces exactly to MinMax through a known defect, and the effect is larger on native surveillance faces (QMUL-SurvFace). INT8 face models should be accepted on FAR and TAR at the deployed operating point.

Keywords: face recognition, post-training quantization, INT8, edge AI, calibration, biometric verification, false accept rate

1 Introduction

Video analytics systems run face recognition on edge accelerators whose fast path is 8-bit integer arithmetic. The usual route is to train in FP32 and apply post-training

quantization (PTQ) with a small calibration set, which fixes the integer range of every activation tensor [1–4]. Compression for deployment is widely surveyed [5], including for biometrics [6]. Toolchains make this one function call, and their defaults are easy to leave unchanged. The converted model is then checked for accuracy loss, typically by comparing identification accuracy (Rank-1) or verification accuracy on a benchmark against the FP32 model [7, 8].

A deployed verifier, however, does not use Rank-1. It compares one similarity score with a fixed threshold that was chosen for a target false accept rate (FAR), usually on the FP32 model, and that threshold sits in the far tail of the impostor score distribution. A quantization error too small to change which gallery identity ranks first can still move that tail. If it does, the system admits more impostors than it was configured for, and nothing in a Rank-1 comparison reveals it.

There is a second, less visible problem. PTQ calibration is computed from a few hundred images, and ONNX Runtime's default calibrator, MinMax, takes each activation range from the most extreme value it sees. Which images happen to be in the calibration set therefore decides the quantized model, and with it the operating point. This paper measures both effects for INT8 PTQ of two widely used face recognizers on low-resolution faces, the setting of surveillance cameras. Our contributions are:

- We quantize ArcFace ResNet-50 with ONNX Runtime's default MinMax calibrator from 10 random calibration sets. All 10 put the FAR at the FP32 threshold above target (1.3–6.0×; sign test $p = 0.001$), also with a held-out threshold, while Rank-1 drops only 2.0–3.5 points. Rank-1 is correlated with the harm across draws, but it compresses a several-fold difference in false accepts into about one point, so a tolerance-based accuracy check passes every one of these models and does not tell how far the operating point moved (Section 4.1).
- Re-deriving the threshold on the INT8 model restores the FAR but costs 18–43% of the TAR at FAR$= 10^{-3}$. Across draws, the FAR excess follows the impostor score mean and spread almost exactly, while genuine scores barely move. Percentile calibration is stable across 10 draws and loses 7–12%; ONNX Runtime's default Entropy setting reduces exactly to MinMax.
- For one calibration draw we replicate the failure on QMUL-SurvFace, a native low-resolution benchmark not used for calibration, compare a second backbone (MobileFaceNet), and localize the loss within the network.
- We give a short acceptance recipe for INT8 face models.

Figure 1 summarizes the study.

2 Related Work

Quantized face recognition

Margin-based recognizers such as ArcFace [9], CosFace [10] and AdaFace [11] are the usual starting point for deployment, and efficient variants are benchmarked in competitions such as EFaR [12]. QuantFace [13] showed that 8-bit quantization-aware training (QAT) of ResNet and MobileFaceNet recognizers is close to lossless on LFW-family accuracy and on IJB-B/C TAR@FAR, with the threshold derived separately

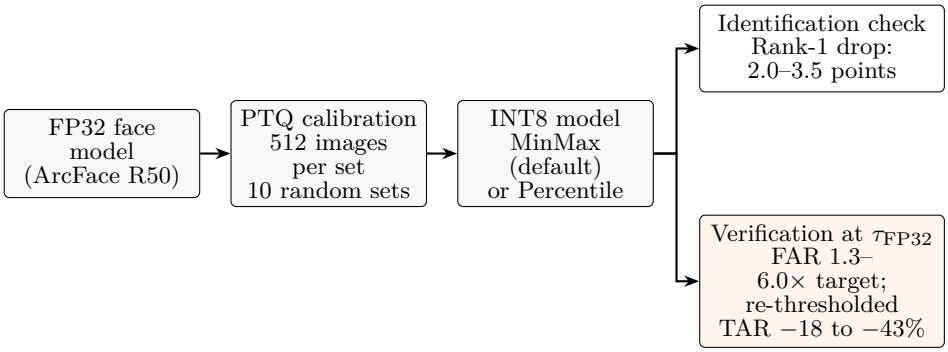


Fig. 1 Study overview. The same INT8 model is judged by the usual identification check and at the deployed verification operating point; numbers are the range over the 10 MinMax calibration sets (R50, TinyFace, 32 px).

for each model. Ef-QuantFace [14], MixQuantBio [15], and recent 6-bit work [16] push to lower bit widths, again with per-model thresholds; Bunda et al. [17] report MobileFaceNet LFW accuracy down to 4 bits. For PTQ, EasyQuant [8] found INT8 MobileFaceNet close to FP32 on several verification sets, and Monteiro et al. [7] compared INT8 FaceNet and TransFace across PyTorch and ONNX using Rank-1 and cosine distance. Neto et al. [18] studied demographic bias after quantization. As far as we can tell from the available texts (several only as abstracts), none of these reports the FAR of the quantized model at the FP32 threshold, compares the standard MinMax, Entropy and Percentile calibrators at a fixed operating point (EasyQuant compares calibration methods by verification accuracy only), or tests low-resolution faces. Hoang and Le [16] report MobileFaceNet as the more sensitive model at 6 bits; at 8 bits we find the opposite.

Calibration and hidden damage

Wu et al. [19] described the max, entropy, and percentile calibrators that current toolchains implement, and compared them on image classification, detection, and language tasks. Which data are used for calibration also matters. Hubara et al. [20] study PTQ with small calibration sets, Williams and Aletras [21] report substantial variation across calibration sets for language models, and ZeroQ [22] replaces real calibration images with synthetic ones. Closest to our draw study, Yuan et al. [23] re-quantize models with many random calibration draws, find MinMax and KL-based calibration the most sensitive to the draw, and report that PTQ can hurt worst-case groups while average accuracy holds. We show what this variation means for face verification: it moves the operating point by a factor that accuracy checks do not reveal. Zhou et al. [24] found that weight-quantized embedding models keep their aggregate metrics while changing many top-1 retrievals, and Leguillier et al. [25] showed that low-bit QAT of speaker-verification models raises impostor scores and counts decision flips at the FP32 threshold. We show the analogue for 8-bit PTQ of face recognizers, tie it to the calibrator, and measure it at a target FAR. Threshold transfer is known to be

Table 1 How quantized face recognition has been evaluated. FAR@τ_{FP32}: FAR of the quantized model at the threshold of the original model. Calib.: comparison of calibration methods; Draws: variation over calibration sets; Low-res: low-resolution faces. †Characterized from the abstract only. "–": not reported, as far as the cited text shows.

Study	Method	Bits	Metrics	FAR@τ	Calib.	Draws	Low-res
QuantFace [13]	QAT	8, 6	LFW-family acc., IJB-B/C TAR@FAR	–	–	–	–
Neto et al. [18]	QAT	–	accuracy by ethnicity	–	–	–	–
Ef-QuantFace [14]	QAT + KD	8, 6	LFW-family acc., IJB-C TPR@FPR	–	–	–	–
MixQuantBio† [15]	mixed prec.	$\leq$8	accuracy	–	–	–	–
Bunda et al.† [17]	QAT	8, 4	LFW accuracy	–	–	–	–
EasyQuant [8]	PTQ	8, 7	verification accuracy	–	✓[a]	–	–
Monteiro et al.† [7]	PTQ	8	Rank-1, cosine distance	–	–	–	–
Hoang and Le [16]	KD vs. PTQ	6	accuracy, TAR@FAR	–	–	–	–
This work	PTQ	8	Rank-1, AUC, FAR@τ_{FP32}, TAR@FAR	✓	✓[b]	✓	✓

[a]TensorRT entropy vs. the proposed scale optimization, by verification accuracy.
[b]MinMax, Entropy (default and 2048-bin), Percentile. TAR@FAR thresholds are re-derived per model in all other rows.

fragile under score-distribution shift [26]; our point is that the quantized model itself is such a shift, and that re-tuning the threshold does not recover the lost accuracy.

Table 1 summarizes how these studies evaluate quantized face recognizers.

3 Experimental Setup

Models

We use the InsightFace ArcFace [9] ResNet-50 (R50, `w600k_r50`) and MobileFaceNet [27] (MBF, `w600k_mbf`) models, both trained on WebFace600K. Each graph is converted to ONNX opset 13, and the converted graph is used for both precisions, so FP32 and INT8 differ only by quantization.

Quantization

We use ONNX Runtime 1.30 static PTQ [28] in QDQ (quantize–dequantize) format with per-channel signed INT8 weights and unsigned INT8 activations, and the library defaults of each calibrator: MinMax (the default), Entropy, and Percentile (99.999th percentile). With its defaults, ONNX Runtime's Entropy calibrator uses 128 histogram bins and 128 quantized bins, which cannot be changed through `quantize_static`. Its Kullback–Leibler (KL) divergence search then has a single candidate, the full

histogram range, which is clipped to the observed minimum and maximum. This defect was reported in 2022 [29]; the issue was closed in October 2026 without a code change, and the defaults are unchanged in the current release (1.31.1). In version 1.30, which we use, on both our models, default Entropy returns exactly the MinMax range for every tensor. We report it as such, and also run a usable Entropy calibration ("Entropy-2048"): ONNX Runtime's own Entropy algorithm with 2048 histogram bins and 128 quantized bins, as in TensorRT [19], with the resulting ranges applied to all activations as overrides. Calibration uses 512 images drawn at random from the TinyFace *training* set; no test image is used. To measure how much the result depends on that draw, we repeat MinMax calibration with 10 different random draws (seeds 0–9) and Percentile with 10 (seeds 0–9; the same seed selects the same images); the detailed analyses in Sections 4.2–5 use draw 0. Inference runs on CPU. One-pass histogram calibration of R50 over 512 images does not fit in memory, so for Entropy and Percentile we collect activations in chunks of 8 images in two passes: the first finds each tensor's global range, the second histograms every chunk on that fixed range and sums the counts. We verified that this is bit-identical to one-pass calibration on a 24-image subset (Section 7). MinMax ranges are reduced chunk by chunk, which is exact.

Data and protocol

TinyFace [30] provides native low-resolution faces (at most 32 px). We use its 3728 test probes and 4443 matched gallery images (2563 identities). Probes are degraded along a ladder of 32, 24, 16, 12, and 8 px by area averaging, as a more distant camera would see them, and resized to the 112 px input; gallery images stay at native resolution. We report closed-set Rank-1 and, for verification, all probe–gallery pairs (genuine pairs share an identity, all others are impostor pairs), which gives 12,308 genuine and 16.6M impostor pairs at each probe size. We use the FAR and TAR terms common in face recognition; they correspond to the false match rate and one minus the false non-match rate of ISO/IEC 19795-1 [31]. The threshold τ is set on FP32 scores at 32 px for FAR$= 10^{-3}$ ($\tau = 0.5035$ for R50, 0.5001 for MBF) and then held fixed, as in a deployed system. Both this threshold and the re-set INT8 thresholds are fitted on the evaluation pairs, so they measure separability, not threshold transfer. Rank-1 is closed-set over the matched gallery, without TinyFace's distractor set, so it is not comparable with published TinyFace numbers. Confidence intervals are 95% bootstrap intervals that resample probe identities (2000 draws), so correlated pairs of one identity are resampled together; thresholds are held at their full-sample values, so the intervals reflect sampling of the genuine pairs and not threshold uncertainty. They describe test-set sampling for a given calibration draw; variation across calibration draws is measured separately (Section 4.1). TinyFace probes are at most 32 px natively, so the 32 px level may involve upsampling of smaller probes. For the second data set we use QMUL-SurvFace [32], native surveillance faces with a median size near 19×25 px, with its official verification protocol (5320 genuine and 5320 impostor pairs) and an extended impostor set of all 50.5M cross-identity pairs, which is needed to estimate FAR at 10^{-3} and below. Calibration for SurvFace is still done on TinyFace training images. Both benchmarks are hard: absolute TAR at FAR$= 10^{-3}$ is low even in FP32

(16.85% for R50 on TinyFace), so throughout we compare precisions of the same model rather than absolute accuracy.

4 Results

4.1 The calibration draw decides how far the operating point moves

Figure 2 shows each calibration draw of R50 as one point (all values in Appendix A). With MinMax, all 10 of the 10 draws put the FAR at the FP32 threshold above the 10^{-3} target (sign test $p = 0.001$), from 1.3 to 6.0 times the target (median 2.2), a 4.5-fold spread between the best and the worst draw. The result does not depend on fitting the threshold on the evaluation pairs: with τ fitted on the FP32 impostor pairs of a random half of the probe identities and the FAR measured on the other half (20 splits), FP32 stays at 1.00 times the target (range 0.83–1.16) while the MinMax draws give 1.3 to 6.1 times. After the threshold is re-set on each INT8 model, TAR at FAR$= 10^{-3}$ is 9.6–13.8% against 16.85% in FP32, a relative loss of 18–43% (median 30%).

Rank-1 over the same draws drops by only 2.0 to 3.5 points from 46.6% (a relative drop of 4–7%), and the area under the ROC curve stays between 0.810 and 0.826. Rank-1 is not blind to the harm: across draws it falls as the FAR rises (Spearman $\rho = -0.84$, $p = 0.002$), and it is lower for every MinMax draw than for every Percentile draw. But it compresses a 4.5-fold difference in false accepts into about 1.4 points of Rank-1: in a paired exact McNemar test on the shared probes, none of the 45 pairs of MinMax draws differs significantly after Bonferroni correction, including the best and the worst draw (FAR ratio 4.5, uncorrected $p = 0.036$). An acceptance check with a tolerance of a few Rank-1 points would pass every MinMax draw, and none of these numbers tells the engineer whether the deployed FAR is 1.3 or 6.0 times the configured target. Percentile calibration is stable across its 10 draws: FAR 1.1–1.3× target, TAR loss 7–12%, and a Rank-1 drop of 0.2–1.6 points.

Across the MinMax draws, the FAR excess follows the impostor scores almost exactly: Spearman $\rho = 0.99$ with the impostor mean and $\rho = 0.99$ with its standard deviation. The draws differ in how far they raise and widen the impostor distribution, which is what moves the tail past τ.

4.2 Anatomy of one draw

The remaining analyses use calibration draw 0, the second-worst of the 10 MinMax draws by FAR. Its MinMax magnitudes are therefore toward the upper end of the range above and should be read as one example, not as typical values; the mechanisms it shows are what we examine. With MinMax calibration, R50 loses 3.5 pp of TinyFace Rank-1 at 32 px (from 46.6%), a 7% relative drop that an acceptance tolerance of a few points would pass. At the FP32 threshold, however, its FAR is 3.9× the target (Table 2). The impostor score mean rises from 0.087 to 0.108 and its standard deviation from 0.099 to 0.110, so more of the impostor tail crosses τ. Default Entropy

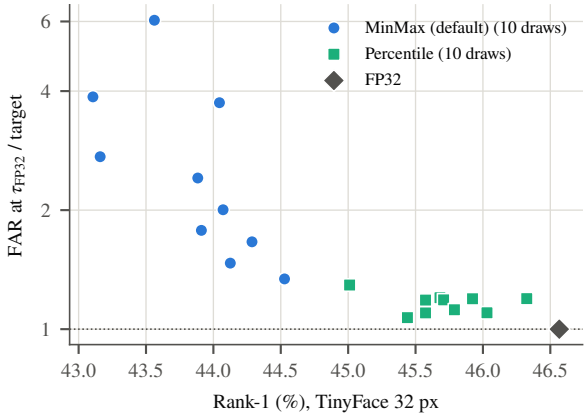


Fig. 2 One point per calibration set of 512 TinyFace training images (R50, TinyFace, 32 px). Every MinMax draw exceeds the FAR target at the FP32 threshold, by 1.3× to 6.0×, while its Rank-1 drop stays within 2.0–3.5 points; Percentile draws stay close to the target.

Table 2 Calibrator comparison on TinyFace (32 px), calibration draw 0. FAR×: INT8 FAR at the FP32 threshold as a multiple of the 10^{-3} target. Imp.: impostor score mean (s.d.). TAR: TAR@FAR= 10^{-3} after re-setting the threshold on the INT8 model's own scores, with 95% CI of the change vs FP32 (pp). R-1: closed-set Rank-1 (%). *ONNX Runtime default Entropy settings; its ranges are identical to MinMax (Sec. 3).

Model	Calib.	R-1	FAR×	Imp.	TAR (%)	ΔTAR CI
R50	FP32	46.6	1.0	0.087 (0.099)	16.85	–
	MinMax	43.1	3.9	0.108 (0.110)	9.75	[-9.2, -5.6]
	Entropy*	43.1	3.9	0.108 (0.110)	9.75	[-9.2, -5.6]
	Entropy-2048	33.0	38.0	0.210 (0.143)	4.62	[-15.8, -9.5]
	Percentile	45.6	1.1	0.091 (0.100)	15.49	[-1.8, -1.0]
MBF	FP32	40.0	1.0	0.129 (0.107)	17.63	–
	MinMax	38.7	1.1	0.136 (0.108)	16.38	[-1.7, -0.9]
	Entropy*	38.7	1.1	0.136 (0.108)	16.38	[-1.7, -0.9]
	Entropy-2048	38.2	1.4	0.151 (0.107)	16.54	[-1.6, -0.6]
	Percentile	39.2	1.0	0.133 (0.107)	17.17	[-0.8, -0.2]

produces the identical model (Section 3), so two of the three ONNX Runtime calibrators we test, as shipped, give this result (a fourth, Distribution, was not tested). Percentile calibration nearly removes it (1.1×, Rank-1 45.6%). Entropy-2048 fails in the opposite direction: it narrows the median activation range to less than half the MinMax width, the impostor mean rises to 0.210, and the FAR is 38.0× target. This failure, however, is visible: Rank-1 falls by 29% relative, to 33.0%. The dangerous case is MinMax, the default, whose damage a Rank-1 check understates. For this draw, MobileFaceNet is much less affected under every calibrator (FAR at most 1.4× target,

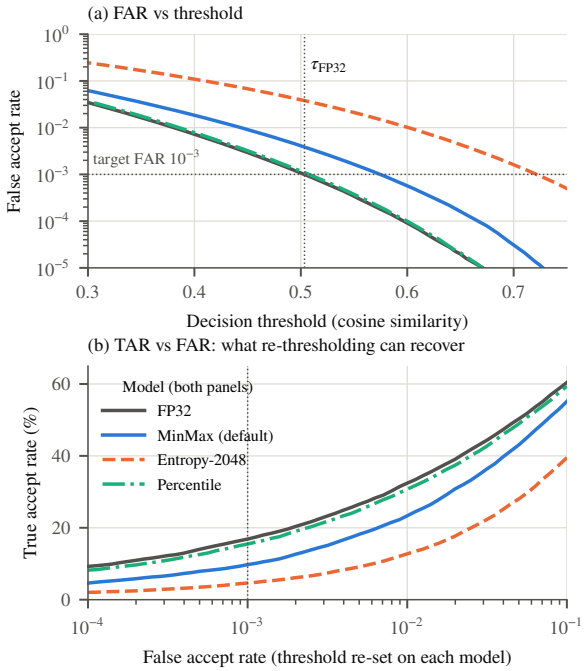


Fig. 3 R50 on TinyFace (32 px, all pairs). (a) FAR as a function of the decision threshold; the dotted lines mark the FP32 threshold τ_{FP32} and the 10^{-3} target. (b) TAR against FAR when the threshold is re-set on each model: the vertical gap at FAR$= 10^{-3}$ is the accuracy that re-thresholding cannot recover.

relative TAR loss at most 7%), which agrees with earlier PTQ results on MobileFaceNet [8]. Fig. 3(a) shows the mechanism directly: the MinMax and Entropy-2048 curves of R50 lie above the FP32 curve at every threshold shown, so at the FP32 threshold both admit more impostors than the target allows; the Percentile curve nearly coincides with FP32.

4.3 Re-tuning the threshold does not recover the loss

A careful operator would re-derive τ on the INT8 model. This restores the FAR by construction, but the INT8 model now accepts fewer genuine pairs at that FAR. For draw 0, R50 TAR@FAR$= 10^{-3}$ falls from 16.85% to 9.75% with MinMax, a relative loss of 42%, and to 4.62% with Entropy-2048 (73%). With Percentile it is 15.49% (8% loss). The quantized model has lost separability between genuine and impostor pairs, not just its calibration, and a threshold cannot buy it back: in Fig. 3(b) the MinMax curve lies below FP32 at every FAR from 10^{-4} to 10^{-1}. For this draw, the relative TAR loss is about 6 times the relative Rank-1 drop. Threshold-free verification summaries understate it just as much: the area under the ROC curve falls only from 0.831 to 0.816 (2% relative) and the best balanced verification accuracy over all thresholds from 76.0% to 74.5%. The damage is concentrated in the low-FAR tail where verifiers

Table 3 TinyFace verification at the FP32 threshold (calibration draw 0; target FAR = 10^{-3}, set at 32 px) as probe faces shrink. ΔFAR is the INT8 excess over FP32 ($\times 10^{-3}$) with 95% identity-cluster bootstrap CI.

Model / calib.	px	FAR FP32	FAR INT8	×target	ΔFAR [95% CI]
R50 MinMax	32	1.00×10^{-3}	3.87×10^{-3}	3.9	2.87 [2.58, 3.17]
	24	9.98×10^{-4}	3.81×10^{-3}	3.8	2.81 [2.52, 3.11]
	16	1.03×10^{-3}	3.84×10^{-3}	3.8	2.81 [2.53, 3.12]
	12	1.20×10^{-3}	5.16×10^{-3}	5.2	3.96 [3.64, 4.31]
	8	1.82×10^{-3}	9.02×10^{-3}	9.0	7.20 [6.77, 7.61]
R50 Percentile	32	1.00×10^{-3}	1.10×10^{-3}	1.1	0.10 [0.06, 0.13]
	24	9.98×10^{-4}	1.08×10^{-3}	1.1	0.09 [0.05, 0.12]
	16	1.03×10^{-3}	1.05×10^{-3}	1.1	0.03 [-0.01, 0.06]
	12	1.20×10^{-3}	1.28×10^{-3}	1.3	0.07 [0.03, 0.12]
	8	1.82×10^{-3}	2.05×10^{-3}	2.0	0.23 [0.17, 0.28]
MBF MinMax	32	1.00×10^{-3}	1.14×10^{-3}	1.1	0.14 [0.11, 0.17]
	24	9.99×10^{-4}	1.14×10^{-3}	1.1	0.14 [0.11, 0.17]
	16	1.22×10^{-3}	1.39×10^{-3}	1.4	0.17 [0.14, 0.21]
	12	1.54×10^{-3}	1.79×10^{-3}	1.8	0.25 [0.20, 0.29]
	8	2.64×10^{-3}	3.19×10^{-3}	3.2	0.55 [0.50, 0.61]

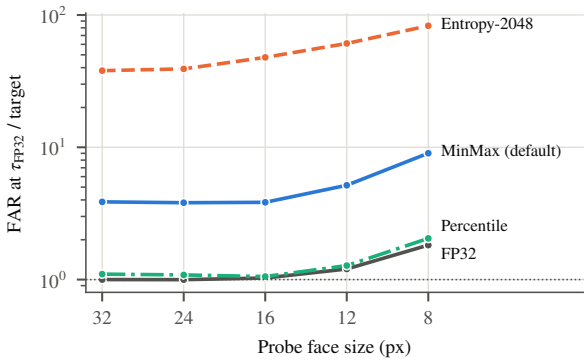


Fig. 4 FAR at the FP32 threshold, as a multiple of the 10^{-3} target, as TinyFace probes shrink (R50). The FP32 curve shows the resolution effect alone; the gap above it is added by INT8.

operate. Genuine scores barely move (mean 0.304 vs. 0.308); it is the shifted and wider impostor distribution that closes the margin.

4.4 Smaller faces

Table 3 and Fig. 4 follow the fixed-threshold FAR as probes shrink (calibration draw 0).

Two effects must be kept apart. Even in FP32 the FAR rises at small sizes (to 1.8×10^{-3} for R50 at 8 px), because τ was set at 32 px. INT8 adds an excess ΔFAR

Table 4 External replication on QMUL-SurvFace (native low-resolution faces; calibration draw 0 on TinyFace training images only). FAR×: extended-impostor FAR at the FP32 threshold for 10^{-3}, as a multiple of target. Off. FAR: official-protocol FAR at the FP32 threshold for 10^{-2}. TAR: TAR@FAR= 10^{-3} after re-thresholding; loss: relative TAR loss vs FP32. Brackets: 95% identity-cluster bootstrap CIs. *Identical to MinMax (default settings).

Model	Calib.	FAR× [CI]	Off. FAR (%)	TAR (%)	Loss % [CI]
R50	FP32	1.0 [0.9, 1.1]	1.0	4.68	–
	MinMax	9.8 [9.4, 10.3]	4.2	1.84	61 [53, 67]
	Entropy*	9.8 [9.4, 10.3]	4.2	1.84	61 [53, 67]
	Entropy-2048	47.2 [45.6, 48.8]	15.1	1.62	65 [58, 73]
	Percentile	1.4 [1.3, 1.5]	1.4	3.65	22 [16, 29]
MBF	FP32	1.0 [0.9, 1.1]	1.0	7.95	–
	MinMax	1.2 [1.1, 1.2]	1.0	7.35	8 [3, 12]
	Entropy*	1.2 [1.1, 1.2]	1.0	7.35	8 [3, 12]
	Entropy-2048	1.2 [1.2, 1.3]	1.2	7.65	4 [-2, 9]
	Percentile	1.2 [1.2, 1.3]	1.2	7.58	5 [0, 9]

on top. For MinMax R50 the excess is flat from 32 to 16 px and grows below that, to 2.5× its 32 px value at 8 px; MBF's much smaller excess grows steadily as faces shrink (4.0× its 32 px value at 8 px). Under Percentile calibration, the excess at 12 px is not larger than at 32 px; only 8 px is. Small faces therefore amplify a shift that the calibrator causes, but they are not its main cause.

4.5 External replication on native surveillance faces

For the same calibration draw, the R50 failure is larger on QMUL-SurvFace (Table 4) than on TinyFace (3.9×): the FAR at the FP32 threshold is 9.8× target with MinMax and 47.2× with Entropy-2048, and on the official protocol a threshold meant for 1% FAR admits 4.2% of impostors. After re-thresholding, TAR@FAR= 10^{-3} falls by 61% relative with MinMax (95% CI [53, 67]%). Percentile reduces but does not remove the loss here (22% relative, CI [16, 29]%), and MBF again loses much less (8% with MinMax, CI [3, 12]%). Absolute TARs are low for all models on this benchmark, which is very hard; the comparison of interest is between precisions of the same model.

5 Where the Loss Arises

Range stretch

MinMax sets each activation range from the extreme values seen during calibration, so rare outliers stretch the range and coarsen the step for typical values. Relative to Percentile, the median R50 activation range is 1.35× wider under MinMax (maximum 3.1×), but MBF shows a similar median stretch (1.33×) without the failure. Range stretch alone does not explain why R50 is sensitive.

Table 5 Localization and ablations on R50 (TinyFace, 32 px, calibration draw 0). Each row starts from MinMax calibration and changes one part. TAR: TAR@FAR= 10^{-3} after re-thresholding. Gap: share of the MinMax→Percentile TAR gap recovered. CI: 95% bootstrap CI of the TAR change vs MinMax (pp).

Variant	Imp.	TAR (%)	Gap (%)	CI
MinMax (all tensors)	0.108	9.75	0	–
5 most-stretched tensors → Percentile	0.103	11.04	23	[+0.9, +1.8]
Stem + stage 1 → Percentile	0.103	11.68	34	[+1.4, +2.6]
Stage 2 → Percentile	0.103	10.93	21	[+0.8, +1.6]
Stage 3 → Percentile	0.096	12.80	53	[+2.3, +4.0]
Stage 4 + head → Percentile	0.108	9.81	1	[-0.1, +0.2]
Embedding layers kept FP32	0.108	9.73	0	[-0.1, +0.0]
Per-tensor weights	0.111	8.95	-14	[-1.2, -0.5]
Percentile (all tensors)	0.091	15.49	100	[+4.5, +7.4]

Outlier tensors

If a few stretched tensors caused the loss, giving only those tensors Percentile ranges would fix it. For the five most stretched R50 tensors (early residual additions and their neighbors), it recovers only 23% of the MinMax-to-Percentile TAR gap (Table 5).

Localization by stage

We repeat the test for each of the four ResNet stages, giving one stage Percentile ranges and keeping MinMax elsewhere (Table 5). The stem with stage 1 recovers 34% of the gap and stage 2 21%, although each holds only about 15% of the quantized tensors; stage 3, the longest, recovers 53%; stage 4 with the embedding head recovers nothing (1%, CI [-0.1, +0.2] pp). Fixed one at a time, the four stages recover 109% of the gap in total, roughly all of it, but finer splits do not combine as simply: the first quarter of the 131 quantized tensors in graph order, which contains all of stage 1, recovers only 17%. We therefore do not claim that the effects add. Splitting the tensors into four equal quarters otherwise gives the same picture as the stages: the second quarter, which spans the end of stage 2 and the start of stage 3, recovers 68%, while the third and fourth quarters recover 4% and 2% when fixed individually. The damage therefore enters in the first half of the network; whether it then propagates or compounds we did not test. Why R50 is more sensitive than MBF despite a similar per-tensor range stretch remains open; error accumulation over R50's greater depth is one candidate that we did not test.

Configuration ablations

Two configuration choices could plausibly cause the shift. Keeping the embedding head (final batch normalization and fully connected layers) in FP32 changes nothing (CI [-0.1, +0.0] pp), and per-tensor weight quantization makes the loss slightly worse (CI [-1.2, -0.5] pp), so for this draw neither the quantized head nor per-channel weights explains the shift.

6 Discussion

What the shift means in deployment

A FAR target is a promise about false alarms. At FAR= 10^{-3}, a verifier should accept about 10 of every 10,000 impostor attempts. With the FP32 threshold left in its configuration, a MinMax INT8 R50 accepts between 13 and 60 (median 22) depending only on which images were used to calibrate it, and Percentile about 11 (Fig. 2, 32 px). In a surveillance deployment that multiplies the alarms an operator must review by an amount nobody measured, while the model's accuracy report shows a Rank-1 drop of 2.0–3.5 points. On native surveillance faces the factor is larger still (Table 4).

A recipe for INT8 face models

(i) Accept the model on verification metrics at the deployed operating point (FAR at the configured threshold and TAR at the target FAR), as recommended for biometric reporting in general [33], never on Rank-1 or accuracy alone. (ii) Compare calibrators instead of taking the library default, and confirm that a calibrator actually changes the ranges: in ONNX Runtime 1.30 the default Entropy setting does not. (iii) Re-derive the threshold on the quantized model, and report the TAR it costs against FP32; if the loss is large, change the calibration, not only the threshold. (iv) Test at the smallest face sizes the camera produces. (v) Where the toolchain fixes the calibrator, steps (i), (iii) and (iv) still apply, and quantization-aware training [13] remains the fallback.

Why ResNet-50 and not MobileFaceNet

For calibration draw 0, both models stretch their activation ranges by a similar median factor under MinMax (Section 5), yet only R50 fails. The loss also enters in the first half of R50. We therefore suspect that depth and the long residual path, rather than any single layer, make R50 sensitive, but we did not test this; a larger study across architectures is the natural next step.

Scope

We test two models from one training recipe, one runtime (ONNX Runtime on CPU), three of its four calibrators, and 8-bit static PTQ. Calibration uses low-resolution TinyFace training images; calibrating on high-quality aligned faces, as many deployments do, may change the size of the effect, and we did not test it. Both benchmarks are low-resolution, so we do not claim the effect for high-quality verification benchmarks such as LFW [34] or IJB-C [35]. Other toolchains (TensorRT, vendor NPU compilers) use other calibrators and defaults, and we do not claim that they fail in the same way; the point is that a Rank-1 check would not tell. We study the calibrators that deployment toolchains expose, not reconstruction-based PTQ methods such as AdaRound [36] or BRECQ [37], which may behave differently. Our verification pairs on TinyFace are our own construction from an identification benchmark, which is why we replicate on the official SurvFace protocol.

7 Threats to Validity

Bounded-memory calibration

Memory-saving shortcuts in calibration are a real risk. In ONNX Runtime 1.30, the option that bounds MinMax memory discards every full chunk without using it, and merging histograms chunk by chunk re-bins them, which moved some Entropy ranges by tens of percent in our tests. We therefore use our own chunked collectors and checked that each reproduces one-pass calibration exactly: all 131 MinMax ranges of R50 on 512 images, and every histogram count and range for Entropy and Percentile on a 24-image subset. We recommend this check to anyone who quantizes under memory limits.

Calibration draws

The draw study (Section 4.1) covers R50 on TinyFace at 32 px, with 10 MinMax and 10 Percentile draws. The size, SurvFace, MobileFaceNet and layer analyses use one draw (draw 0, the second-worst MinMax draw), so their MinMax magnitudes are for that draw; their bootstrap intervals cover test-set sampling, not calibration sampling. We did not vary the size or the source of the calibration set.

Dependent pairs

Pairs that share an image are not independent; we therefore resample identities, not pairs, for all intervals.

8 Conclusion

With ONNX Runtime's default MinMax calibrator, INT8 quantization of a face recognizer puts its verification operating point above the target in every calibration draw we tried, by an amount that ranges from 1.3 to 6.0 times and is set by which images happened to be used for calibration. Rank-1 and AUC fall by only a few points and do not reveal how far the operating point moved, and re-tuning the threshold restores the FAR only at the cost of a large share of true accepts. Percentile calibration is far more stable and close to FP32 on TinyFace, although for the draw we tested on SurvFace it still loses about a fifth of the TAR. Verification at the deployed operating point should be part of every acceptance test for quantized face models, and the calibrator should be chosen, not inherited.

Appendix A All calibration draws

Table A1 lists every calibration draw summarized in Section 4.1 and Fig. 2.

Table A1 Every calibration draw of R50 (TinyFace, 32 px; 512 training images per draw; the same seed selects the same images for both calibrators). FAR×: FAR at the FP32 threshold as a multiple of the 10^{-3} target. TAR: TAR@FAR= 10^{-3} after re-thresholding. FP32: Rank-1 46.6%, TAR 16.85%, AUC 0.831.

	MinMax (default)				Percentile			
Draw	FAR×	TAR (%)	Rank-1 (%)	AUC	FAR×	TAR (%)	Rank-1 (%)	AUC
0	3.87	9.75	43.1	0.816	1.10	15.49	45.6	0.828
1	1.47	13.18	44.1	0.820	1.19	15.26	45.7	0.827
2	1.34	13.81	44.5	0.826	1.19	14.88	45.9	0.826
3	1.78	12.62	43.9	0.819	1.07	15.62	45.4	0.826
4	2.41	11.49	43.9	0.819	1.19	15.19	46.3	0.827
5	1.66	12.95	44.3	0.821	1.12	15.57	45.8	0.827
6	6.04	9.64	43.6	0.810	1.29	14.75	45.0	0.824
7	2.00	12.11	44.1	0.821	1.20	15.11	45.7	0.827
8	3.74	11.33	44.0	0.817	1.18	15.20	45.6	0.826
9	2.73	11.24	43.2	0.814	1.10	15.69	46.0	0.826

Declarations

Funding: no funding was received. **Competing interests:** the authors declare none. **Ethics approval and consent:** not applicable; the study uses only public, previously published face data sets (TinyFace, QMUL-SurvFace) under their research licenses, and no new data were collected. **Data availability:** TinyFace and QMUL-SurvFace are public research data sets; the models are public InsightFace releases (non-commercial research license). **Code availability:** all scripts, per-pair results and the numbers generator used for this paper will be released in a public repository on publication. **Use of AI tools:** an AI assistant (Claude, Anthropic) was used to assist with code development, literature search, and the drafting and editing of the text. The study design, analysis decisions and conclusions are the authors' own; the authors verified all code, results and text and take full responsibility for the content. **Author contributions:** M.R.T.
designed the study, ran the experiments and wrote the manuscript; M.H.A.
reviewed the design and the manuscript.

References

[1] B. Jacob et al., "Quantization and training of neural networks for efficient integer-arithmetic-only inference," in *Proc. CVPR*, 2018, pp. 2704–2713, doi:10.1109/CVPR.2018.00286.

[2] R. Krishnamoorthi, "Quantizing deep convolutional networks for efficient inference: A whitepaper," arXiv:1806.08342, 2018.

[3] M. Nagel, M. Fournarakis, R. A. Amjad, Y. Bondarenko, M. van Baalen, and T. Blankevoort, "A white paper on neural network quantization," arXiv:2106.08295, 2021.

[4] A. Gholami, S. Kim, Z. Dong, Z. Yao, M. W. Mahoney, and K. Keutzer, "A survey of quantization methods for efficient neural network inference," arXiv:2103.13630, 2021.

[5] L. Deng, G. Li, S. Han, L. Shi, and Y. Xie, "Model compression and hardware acceleration for neural networks: A comprehensive survey," *Proc. IEEE*, vol. 108, no. 4, pp. 485–532, 2020, doi:10.1109/JPROC.2020.2976475.

[6] E. Caldeira, P. C. Neto, M. Huber, N. Damer, and A. F. Sequeira, "Model compression techniques in biometrics applications: A survey," arXiv:2401.10139, 2024.

[7] C. H. Monteiro, E. Raphaloski, and E. T. Matsubara, "Experimental evaluation of quantization methods in facial recognition," in *Anais Estendidos do SBAC-PAD (LeanDL-HPC)*, 2025, pp. 108–115.

[8] D. Wu, Q. Tang, Y. Zhao, M. Zhang, Y. Fu, and D. Zhang, "EasyQuant: Post-training quantization via scale optimization," arXiv:2006.16669, 2020.

[9] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive angular margin loss for deep face recognition," in *Proc. CVPR*, 2019, pp. 4690–4699.

[10] H. Wang et al., "CosFace: Large margin cosine loss for deep face recognition," in *Proc. CVPR*, 2018.

[11] M. Kim, A. K. Jain, and X. Liu, "AdaFace: Quality adaptive margin for face recognition," in *Proc. CVPR*, 2022.

[12] J. N. Kolf et al., "EFaR 2023: Efficient face recognition competition," arXiv:2308.04168, 2023.

[13] F. Boutros, N. Damer, and A. Kuijper, "QuantFace: Towards lightweight face recognition by synthetic data low-bit quantization," in *Proc. ICPR*, 2022, pp. 855–862, doi:10.1109/ICPR56361.2022.9955645.

[14] W. Gazali, J. M. Kho, J. Santoso, and Williem, "Ef-QuantFace: Streamlined face recognition with small data and low-bit precision," arXiv:2402.18163, 2024.

[15] J. N. Kolf, J. Elliesen, N. Damer, and F. Boutros, "MixQuantBio: Towards extreme face and periocular recognition model compression with mixed-precision quantization," *Eng. Appl. Artif. Intell.*, vol. 137, Art. no. 109114, Nov. 2024, doi:10.1016/j.engappai.2024.109114.

[16] K. T. Hoang and C.-Q. Le, "Fast-converging and architecture-agnostic 6-bit face recognition via gradient coordination and refined data," in *Proc. MAPR*, 2026.

[17] S. Bunda, L. Spreeuwers, and C. Zeinstra, "Sub-byte quantization of mobile face recognition convolutional neural networks," in *Proc. BIOSIG*, 2022,

doi:10.1109/BIOSIG55365.2022.9897025.

[18] P. C. Neto, E. Caldeira, J. S. Cardoso, and A. F. Sequeira, "Compressed models decompress race biases: What quantized models forget for fair face recognition," in *Proc. BIOSIG*, 2023.

[19] H. Wu, P. Judd, X. Zhang, M. Isaev, and P. Micikevicius, "Integer quantization for deep learning inference: Principles and empirical evaluation," arXiv:2004.09602, 2020.

[20] I. Hubara, Y. Nahshan, Y. Hanani, R. Banner, and D. Soudry, "Accurate post training quantization with small calibration sets," in *Proc. ICML*, PMLR vol. 139, 2021 (arXiv:2006.10518).

[21] M. Williams and N. Aletras, "On the impact of calibration data in post-training quantization and pruning," in *Proc. ACL*, 2024 (arXiv:2311.09755).

[22] Y. Cai, Z. Yao, Z. Dong, A. Gholami, M. W. Mahoney, and K. Keutzer, "ZeroQ: A novel zero shot quantization framework," in *Proc. CVPR*, 2020.

[23] Z. Yuan et al., "Benchmarking the reliability of post-training quantization: A particular focus on worst-case performance," arXiv:2303.13003, 2023.

[24] L. Zhou, A. Zirilli, D. Solombrino, R. Dessì, and E. Rodolà, "The undetected damage of quantization on retrieval and how to fix it," arXiv:2609.24322, 2026.

[25] H. Leguillier, D. Matrouf, G. Lechien, and M. Rouvier, "On low-bit quantization errors in speaker verification: Diagnostic and mitigation," in *Proc. Odyssey: Speaker Lang. Recognit. Workshop*, 2026, arXiv:2606.08078.

[26] A. Apila, "On the use of synthetic data for threshold calibration in face recognition: Performance and security implications for border control systems," in *Proc. IJCB*, 2026, arXiv:2607.25990.

[27] S. Chen, Y. Liu, X. Gao, and Z. Han, "MobileFaceNets: Efficient CNNs for accurate real-time face verification on mobile devices," in *Proc. CCBR*, 2018, pp. 428–438, doi:10.1007/978-3-319-97909-0_46.

[28] ONNX Runtime developers, "ONNX Runtime," https://onnxruntime.ai, version 1.30, 2026.

[29] ONNX Runtime GitHub issue #10261, "quantization tools: change num_bins default value," Jan. 2022. [Online]. Available: https://github.com/microsoft/onnxruntime/issues/10261

[30] Z. Cheng, X. Zhu, and S. Gong, "Low-resolution face recognition," in *Proc. ACCV*, 2018, pp. 605–621, doi:10.1007/978-3-030-20893-6_38.

[31] ISO/IEC 19795-1:2021, "Information technology – Biometric performance testing and reporting – Part 1: Principles and framework," International Organization for Standardization, 2021.

[32] Z. Cheng, X. Zhu, and S. Gong, "Surveillance face recognition challenge," arXiv:1804.09691, 2018.

[33] A. Ahmed and M. H. Imtiaz, "Beyond ROC-AUC: Operating-point performance reporting for biometric verification," arXiv:2606.20680, 2026.

[34] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Univ. of Massachusetts, Amherst, Tech. Rep. 07-49, 2007.

[35] B. Maze et al., "IARPA Janus Benchmark-C: Face dataset and protocol," in *Proc. ICB*, 2018, pp. 158–165, doi:10.1109/ICB2018.2018.00033.

[36] M. Nagel, R. A. Amjad, M. van Baalen, C. Louizos, and T. Blankevoort, "Up or down? Adaptive rounding for post-training quantization," in *Proc. ICML*, 2020, pp. 7197–7206.

[37] Y. Li et al., "BRECQ: Pushing the limit of post-training quantization by block reconstruction," in *Proc. ICLR*, 2021.