MedMNIST-EdgeAI — Phase-1 (HAM10000)

Generated: 2025-10-20 17:11:16

Repository root: D:\MedMNIST-EdgeAIv2

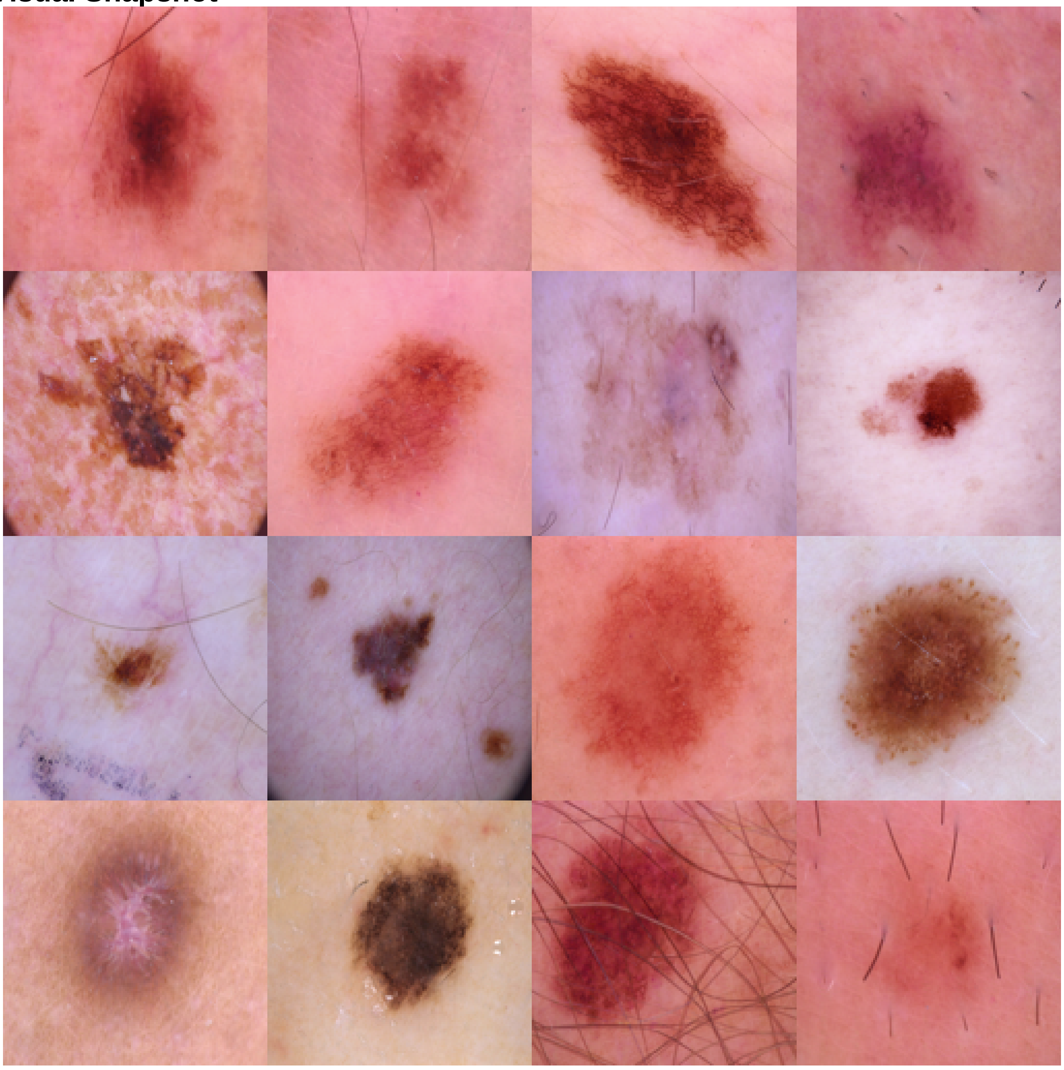
This report consolidates teacher/student performance, calibration, operating points, robustness under corruptions, and efficiency (latency/memory). It now also embeds TensorBoard scalar plots and a scalar summary table pulled from your .tfevents.

1. Teacher & Student Roles

Teacher: high-capacity ResNet-50 trained on HAM10000 at native resolution; provides soft targets for knowledge distillation.

Students: ResNet-18, MobileNetV2, EfficientNet-B0 distilled with KD (T, α) and optional Attention Transfer (λ), evaluated over multiple seeds.

Dataset Visual Snapshot



2. Performance (Test-set with CI)

Teacher summary:

model	accuracy_mean	accuracy_lo	accuracy_hi	macro_f1_mean	macro_f1_lo	macro_f1_hi
runs_ham10000_resnet50	0.893	0.8796804792810784	0.9071392910634049	0.834	0.8041329541329876	0.8605454187329692

Students summary:

model	accuracy_mean	accuracy_lo	accuracy_hi	macro_f1_mean	macro_f1_lo	macro_f1_hi
distilled_efficientnetb0_ham10000	0.81	0.7929016966067864	0.8273453093812375	0.656	0.6143198532359181	0.6924848902971202
distilled_mobilenetv2_ham10000	0.784	0.7659680638722555	0.8023952095808383	0.541	0.5098655666558318	0.5691133356818394
distilled_resnet18_ham10000	0.852	0.8363273453093812	0.8672779441117764	0.755	0.7153264067221303	0.786634769600519

Best student by macro_f1_mean: **distilled_resnet18_ham10000** = 0.755.

3. Calibration & Operating Points

Calibration metrics:

model	seed	accuracy	nll	brier	ece_uniform	ece_adaptive
distilled_efficientnetb0_ham10000	seed_0	0.810379241516966	381.732	0.379	0.1894070034969352	0.189
distilled_mobilenetv2_ham10000	seed_0	0.7839321357285429	484.922	0.432	0.2160592079162597	0.216
distilled_resnet18_ham10000	seed_0	0.8517964071856288	312.122	0.296	0.1481770277023315	0.148

Operating points (per model/seed):

model	seed	tau	macro_f1_opt
distilled_efficientnetb0_ham10000	seed_0	0.0	0.656
distilled_mobilenetv2_ham10000	seed_0	0.0	0.541
distilled_resnet18_ham10000	seed_0	0.0	0.755

4. Robustness under Corruptions

Macro-F1 across corruption levels:

acc	macro_f1	tag	model	seed
0.810379241516966	0.656	gaussian_0.1	distilled_efficientnetb0_ham10000	seed_0
0.810379241516966	0.656	gaussian_0.2	distilled_efficientnetb0_ham10000	seed_0
0.811377245508982	0.658	gaussian_0.3	distilled_efficientnetb0_ham10000	seed_0
0.8158682634730539	0.663	jpeg_90	distilled_efficientnetb0_ham10000	seed_0
0.8133732534930139	0.654	jpeg_70	distilled_efficientnetb0_ham10000	seed_0
0.8158682634730539	0.659	jpeg_50	distilled_efficientnetb0_ham10000	seed_0
0.779441117764471	0.584	contrast_0.8	distilled_efficientnetb0_ham10000	seed_0
0.659181636726547	0.345	contrast_0.6	distilled_efficientnetb0_ham10000	seed_0
0.7839321357285429	0.541	gaussian_0.1	distilled_mobilenetv2_ham10000	seed_0
0.7829341317365269	0.54	gaussian_0.2	distilled_mobilenetv2_ham10000	seed_0
0.7819361277445109	0.536	gaussian_0.3	distilled_mobilenetv2_ham10000	seed_0
0.7869261477045908	0.545	jpeg_90	distilled_mobilenetv2_ham10000	seed_0
0.782435129740519	0.536	jpeg_70	distilled_mobilenetv2_ham10000	seed_0
0.7849301397205589	0.543	jpeg_50	distilled_mobilenetv2_ham10000	seed_0
0.7604790419161677	0.483	contrast_0.8	distilled_mobilenetv2_ham10000	seed_0
0.6402195608782435	0.27	contrast_0.6	distilled_mobilenetv2_ham10000	seed_0
0.8517964071856288	0.755	gaussian_0.1	distilled_resnet18_ham10000	seed_0
0.8517964071856288	0.755	gaussian_0.2	distilled_resnet18_ham10000	seed_0
0.8512974051896207	0.754	gaussian_0.3	distilled_resnet18_ham10000	seed_0
0.8552894211576846	0.758	jpeg_90	distilled_resnet18_ham10000	seed_0
0.8527944111776448	0.747	jpeg_70	distilled_resnet18_ham10000	seed_0
0.8507984031936128	0.746	jpeg_50	distilled_resnet18_ham10000	seed_0
0.81187624750499	0.663	contrast_0.8	distilled_resnet18_ham10000	seed_0
0.719560878243513	0.451	contrast_0.6	distilled_resnet18_ham10000	seed_0

5. Efficiency (Latency & Memory)

GPU latency (aggregated):

ckpt	model	seed	device	batch	imgsz	lat_ms_mean	lat_ms_std	lat_ms_p50	lat_ms_p90	lat_ms_p99	throughput_fps
D:\MedMNIST-EdgeAIv2\models\students\distilled_efficientnet…	distilled_efficientnetb0_ham10000	seed_0	cuda	1	224	4.8	0.148	4.8	5.010360019514337	5.195631969836541	206.92272391257524
D:\MedMNIST-EdgeAIv2\models\students\distilled_mobilenetv2_…	distilled_mobilenetv2_ham10000	seed_0	cuda	1	224	3.3	0.061	3.3	3.3566700061783195	3.40099205088336	305.15249995875126
D:\MedMNIST-EdgeAIv2\models\students\distilled_resnet18_ham…	distilled_resnet18_ham10000	seed_0	cuda	1	224	2.3	0.181	2.2	2.575390046695248	2.590343989431858	443.29444004730254

CPU latency (aggregated):

ckpt	model	seed	device	batch	imgsz	lat_ms_mean	lat_ms_std	lat_ms_p50	lat_ms_p90	lat_ms_p99	throughput_fps
D:\MedMNIST-EdgeAIv2\models\students\distilled_efficientnet…	distilled_efficientnetb0_ham10000	seed_0	cpu	1	224	34.9	0.953	34.8	35.69034003303386	38.579854981508106	28.628836750156
D:\MedMNIST-EdgeAIv2\models\students\distilled_mobilenetv2_…	distilled_mobilenetv2_ham10000	seed_0	cpu	1	224	24.1	1.203	23.8	25.95014002872631	26.73692700045649	41.40883027376206
D:\MedMNIST-EdgeAIv2\models\students\distilled_resnet18_ham…	distilled_resnet18_ham10000	seed_0	cpu	1	224	23.0	0.307	23.0	23.322999995434657	24.186682010185898	43.39109758824055

Model memory footprint:

ckpt	model	seed	params_bytes	params_mib	peak_cuda_bytes	peak_cuda_mib	img sz	batch
D:\MedMNIST-EdgeAIv2\models\students\distilled_efficientnet…	distilled_efficientnetb0_ham10000	seed_0	16234516	15.5	26550272	25.3	224	1
D:\MedMNIST-EdgeAIv2\models\students\distilled_mobilenetv2_…	distilled_mobilenetv2_ham10000	seed_0	9068220	8.6	19385344	18.5	224	1
D:\MedMNIST-EdgeAIv2\models\students\distilled_resnet18_ham…	distilled_resnet18_ham10000	seed_0	44758972	42.7	55447552	52.9	224	1

6. Pareto (Accuracy vs Latency)

model	score	latency_ms	params_mib	peak_cuda_mib
distilled_resnet18_ham10000	0.755	2.2	42.7	52.9
distilled_mobilenetv2_ham10000	0.541	3.3	8.6	18.5
distilled_efficientnetb0_ham10000	0.656	4.8	15.5	25.3

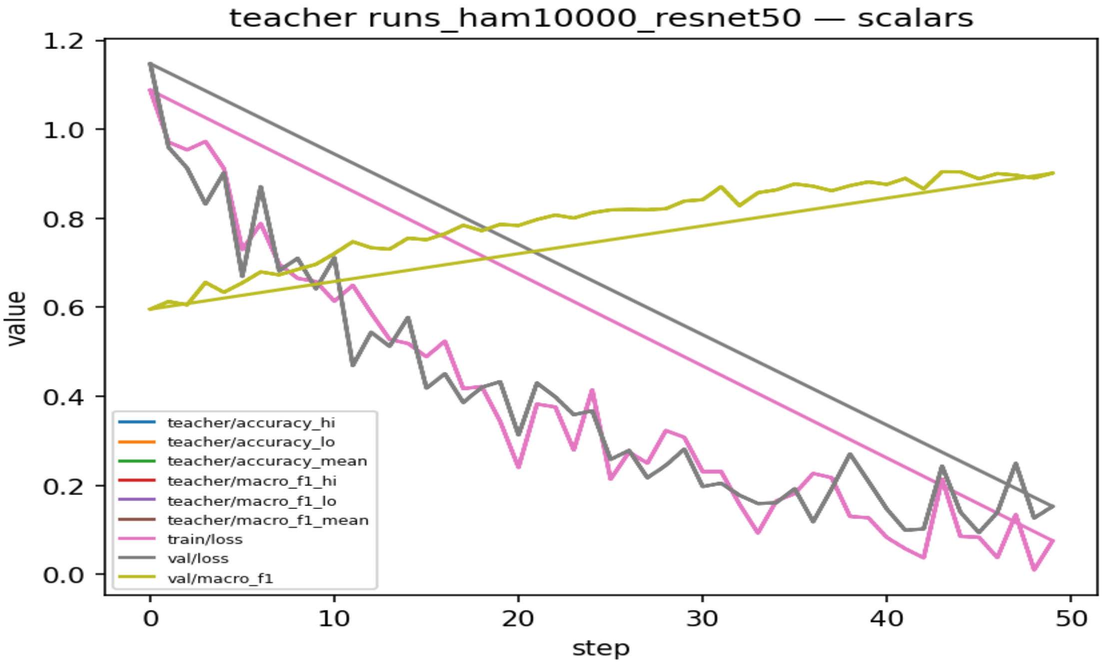
See appended figure: Acc-vs-Latency bubble plot.

7. TensorBoard Scalars & Images

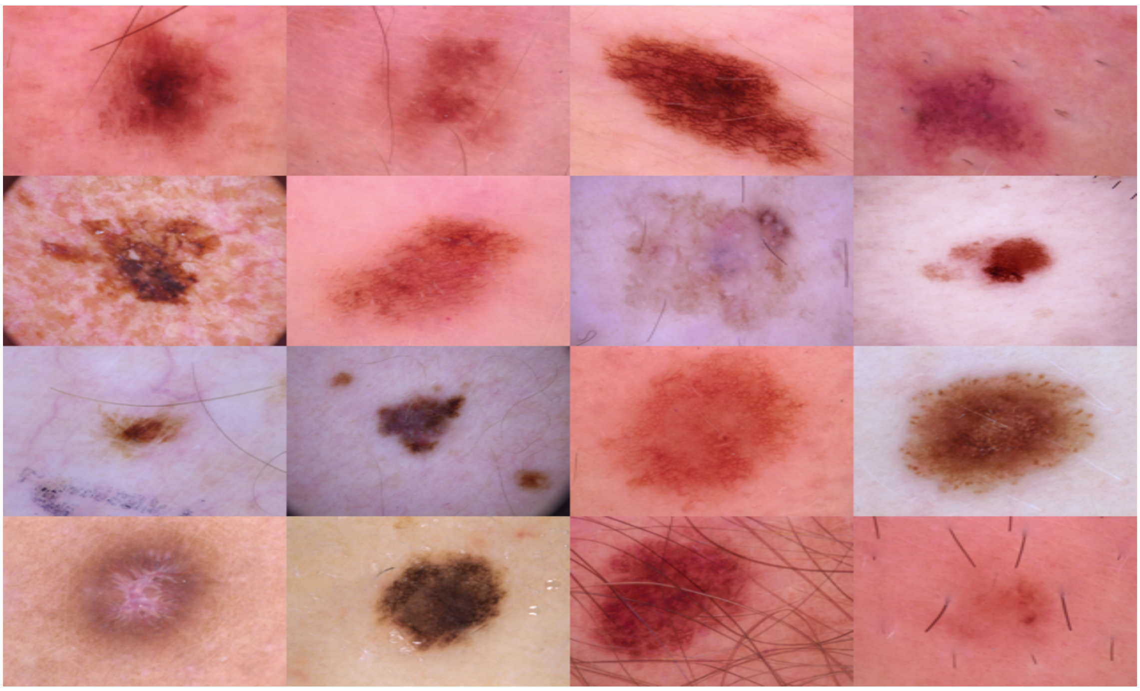
TensorBoard scalar summary (last/best values):

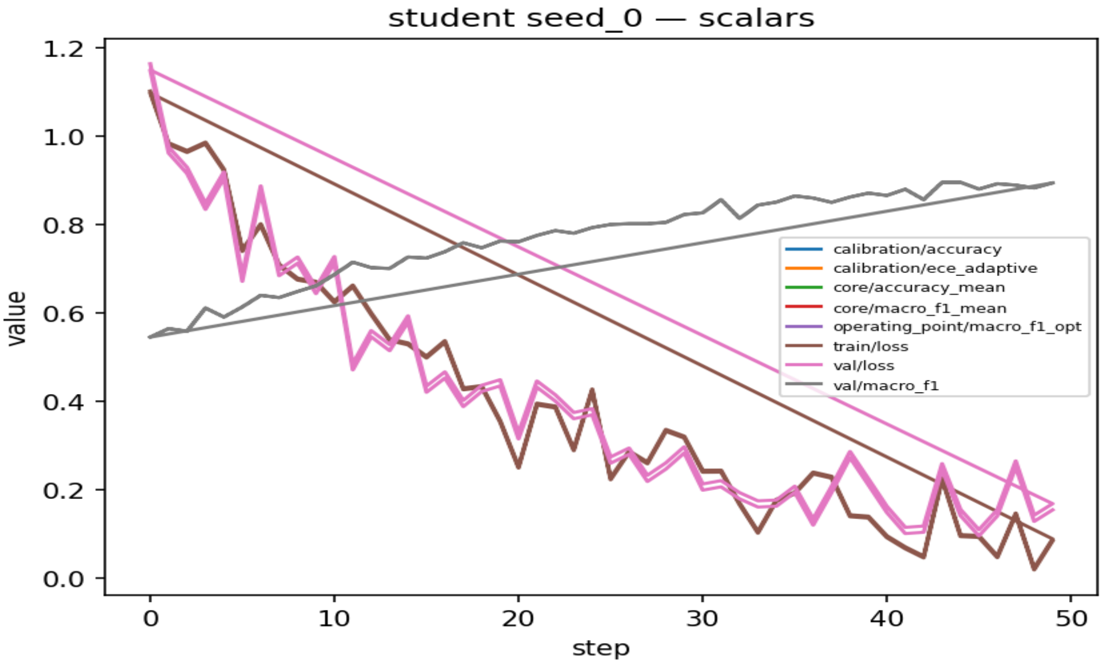
run	tag	last	best	best_step
student\\seed_0	calibration\\accuracy	0.8104	0.8104	0
student\\seed_0	calibration\\accuracy	0.7839	0.7839	0
student\\seed_0	calibration\\accuracy	0.8518	0.8518	0
student\\seed_0	calibration\\ece_adaptive	0.1894	0.1894	0
student\\seed_0	calibration\\ece_adaptive	0.2161	0.2161	0
student\\seed_0	calibration\\ece_adaptive	0.1482	0.1482	0
student\\seed_0	core\\accuracy_mean	0.8104	0.8104	0
student\\seed_0	core\\accuracy_mean	0.7839	0.7839	0
student\\seed_0	core\\accuracy_mean	0.8518	0.8518	0
student\\seed_0	core\\macro_f1_mean	0.6558	0.6558	0
student\\seed_0	core\\macro_f1_mean	0.5411	0.5411	0
student\\seed_0	core\\macro_f1_mean	0.7545	0.7545	0
student\\seed_0	operating_point\\macro_f1_opt	0.6558	0.6558	0
student\\seed_0	operating_point\\macro_f1_opt	0.5411	0.5411	0
student\\seed_0	operating_point\\macro_f1_opt	0.7545	0.7545	0
student\\seed_0	train\\loss	0.0902	0.0158	48
student\\seed_0	train\\loss	0.0761	0.0105	48
student\\seed_0	train\\loss	0.0852	0.0195	48
student\\seed_0	val\\loss	0.171	0.1048	45
student\\seed_0	val\\loss	0.1674	0.0972	45
student\\seed_0	val\\loss	0.1553	0.0964	45
student\\seed_0	val\\macro_f1	0.8945	0.8959	44
student\\seed_0	val\\macro_f1	0.8945	0.8959	44
student\\seed_0	val\\macro_f1	0.8945	0.8959	44
teacher\\runs_ham10000_resnet50	teacher\\accuracy_hi	0.9071	0.9071	0
teacher\\runs_ham10000_resnet50	teacher\\accuracy_lo	0.8797	0.8797	0
teacher\\runs_ham10000_resnet50	teacher\\accuracy_mean	0.8932	0.8932	0
teacher\\runs_ham10000_resnet50	teacher\\macro_f1_hi	0.8605	0.8605	0
teacher\\runs_ham10000_resnet50	teacher\\macro_f1_lo	0.8041	0.8041	0
teacher\\runs_ham10000_resnet50	teacher\\macro_f1_mean	0.8341	0.8341	0
teacher\\runs_ham10000_resnet50	train\\loss	0.0757	0.01	48
teacher\\runs_ham10000_resnet50	val\\loss	0.1528	0.0939	45
teacher\\runs_ham10000_resnet50	val\\macro_f1	0.9016	0.9046	43

TensorBoard plots/images:

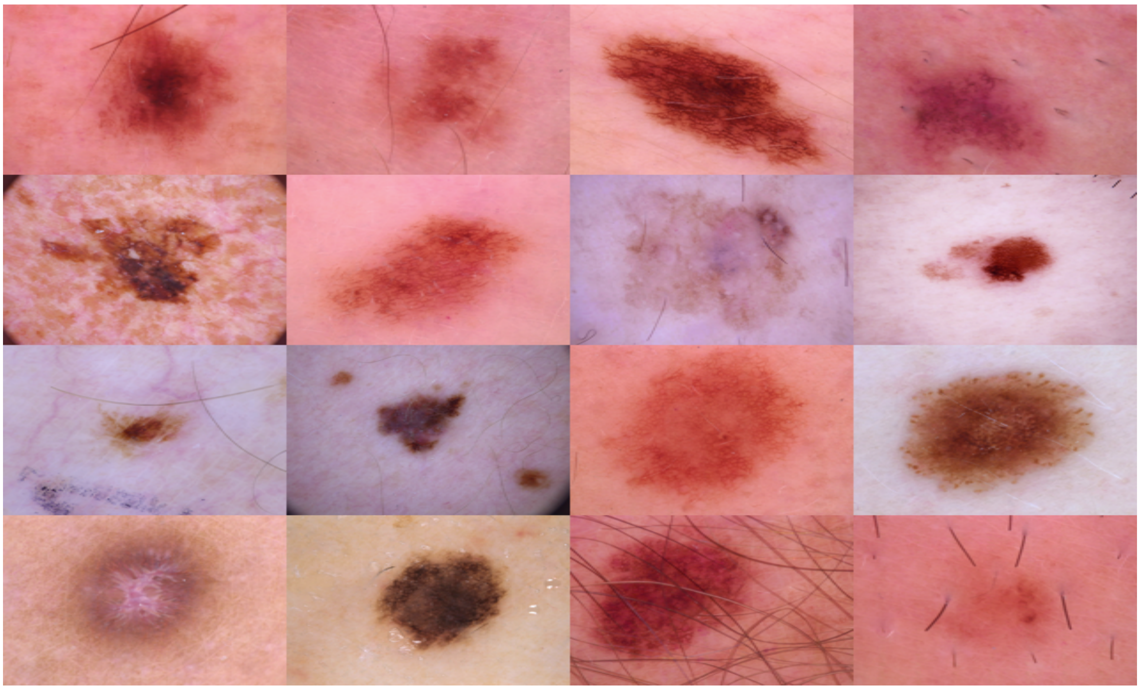


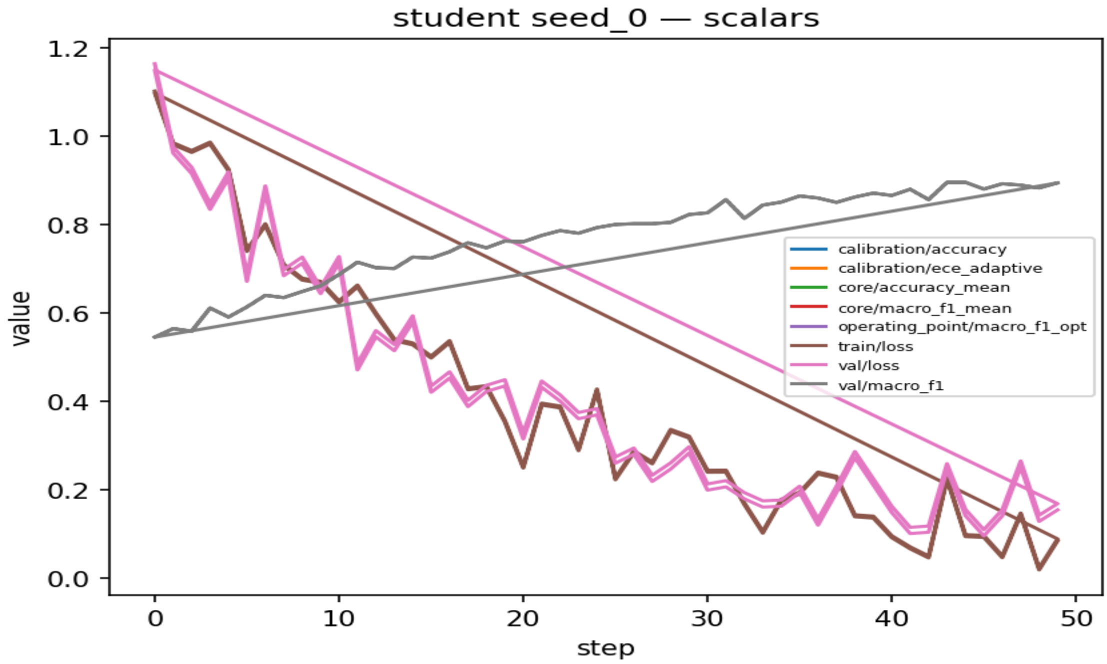
teacher runs_ham10000_resnet50 — scalars
value
step
teacher/accuracy_hi
teacher/accuracy_lo
teacher/accuracy_mean
teacher/macro_f1_hi
teacher/macro_f1_lo
teacher/macro_f1_mean
train/loss
val/loss
val/macro_f1



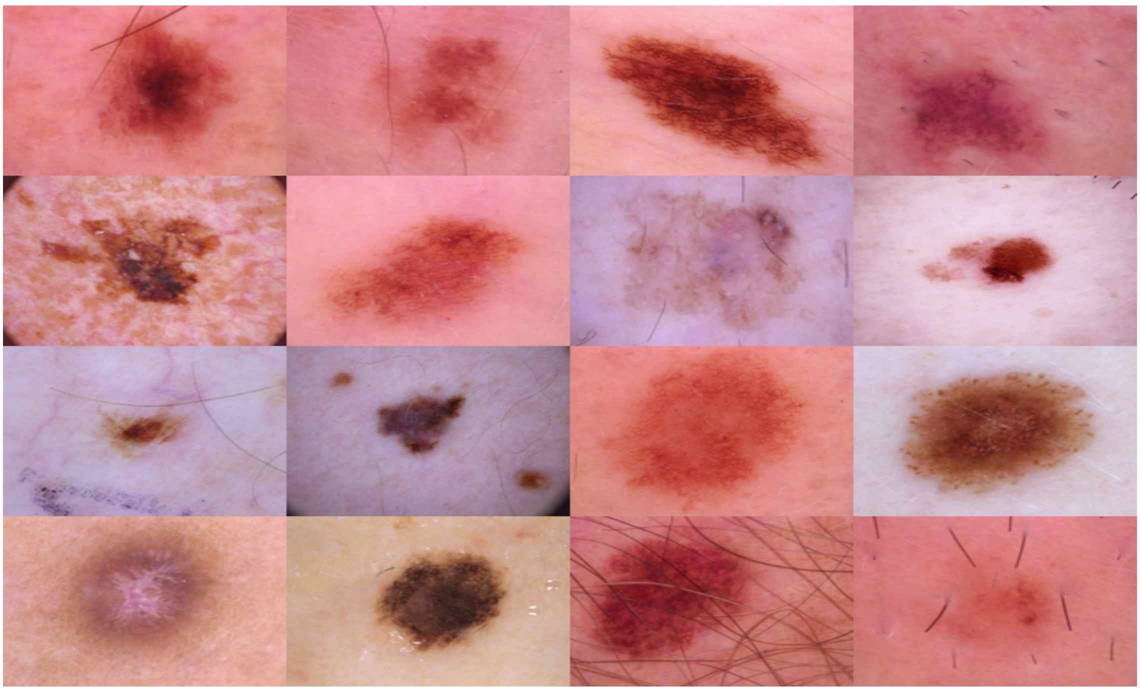


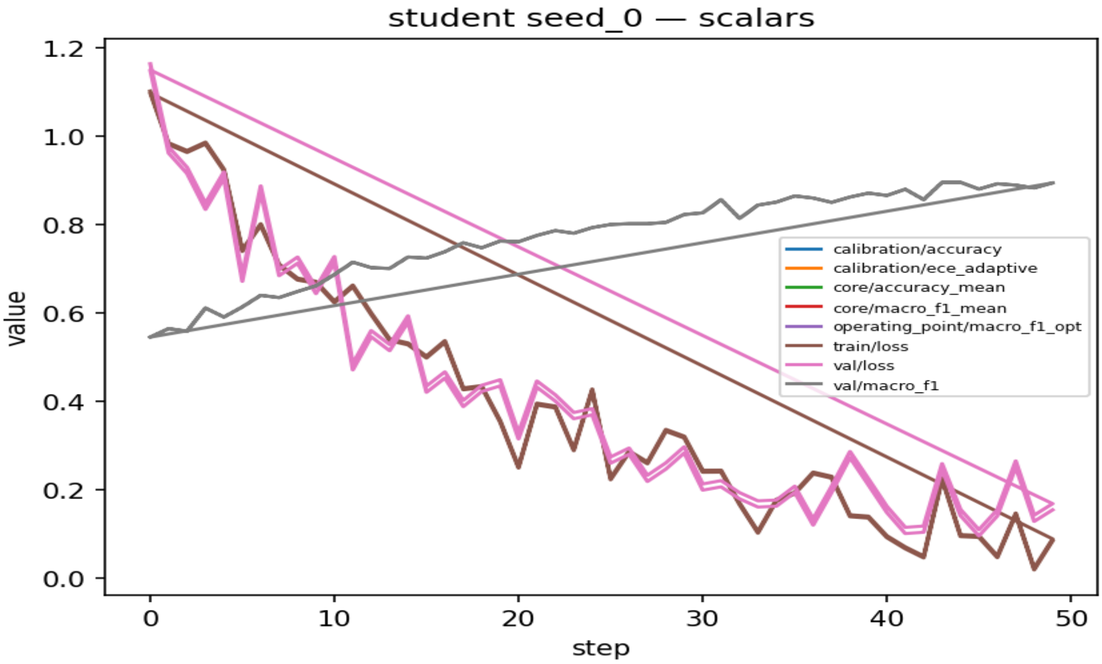
student seed_0 — scalars
value
step
0.0
0.2
0.4
0.6
0.8
1.0
1.2
0
10
20
30
40
50
calibration/accuracy
calibration/ece_adaptive
core/accuracy_mean
core/macro_f1_mean
operating_point/macro_f1_opt
train/loss
val/loss
val/macro_f1

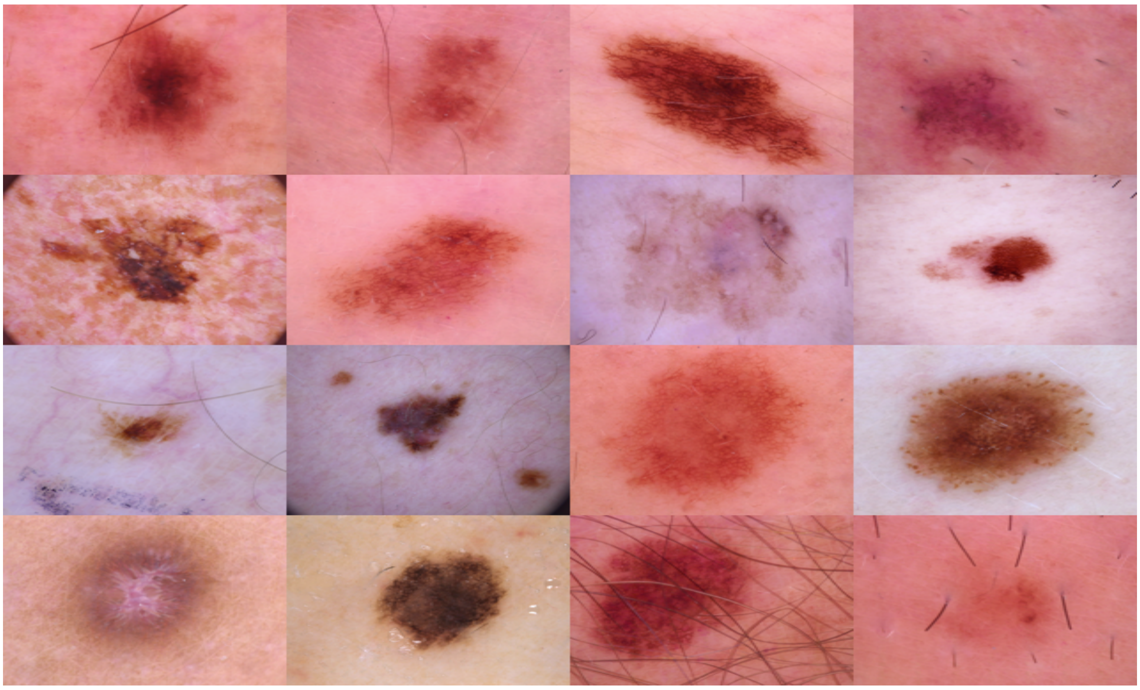




student seed_0 — scalars
1.2
1.0
0.8
0.6
0.4
0.2
0.0
value
0
10
20
30
40
50
step
calibration/accuracy
calibration/ece_adaptive
core/accuracy_mean
core/macro_f1_mean
operating_point/macro_f1_opt
train/loss
val/loss
val/macro_f1




student seed_0 — scalars
value
step
calibration/accuracy
calibration/ece_adaptive
core/accuracy_mean
core/macro_f1_mean
operating_point/macro_f1_opt
train/loss
val/loss
val/macro_f1



8. Supplementary Notes from DOCX

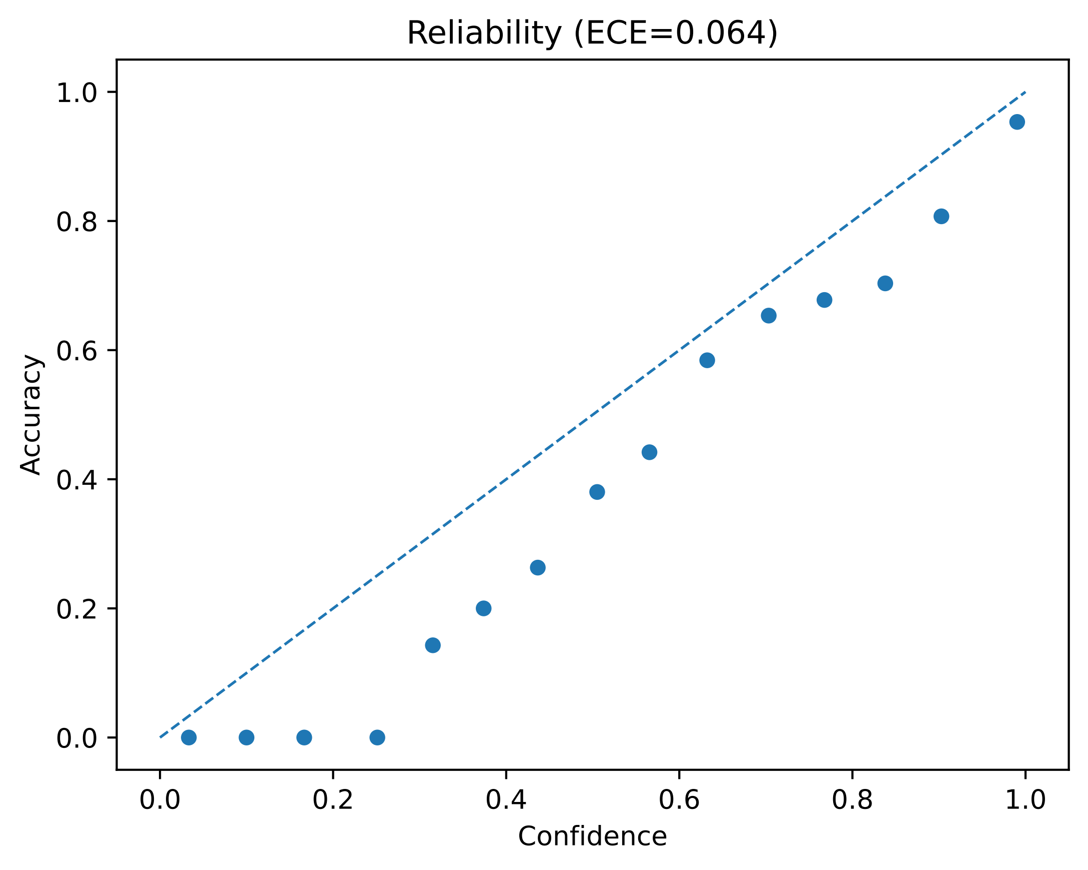
- ResNet50vsResNet18 Beta-1000.docx

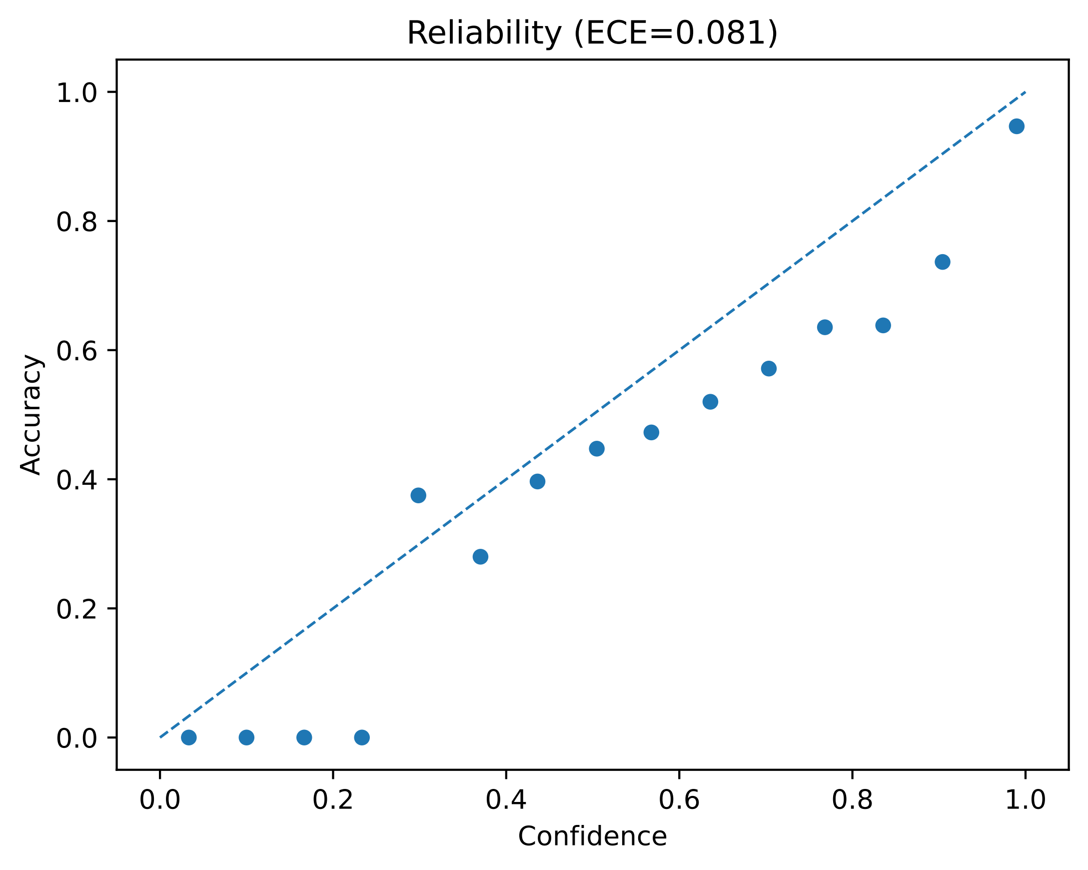
This is exactly the kind of crisp comparative data that reviewers love — you've got a clear quantitative teacher–student profile, per-class confusion matrices, and full metrics. Let's analyze it like a researcher. ■ Macro-level summary That's roughly 84–90% retention of the teacher's performance with ~55% fewer parameters and significantly lower inference cost — a good KD outcome on HAM10000, where minority classes dominate macro-F1 sensitivity. ■ Per-class analysis Interpretation: The student model faithfully inherited teacher knowledge for major and well-separated classes (5, 6), but compressed representation hurts fine-grained or underrepresented lesions (0, 2, 3, 4). That's expected: ResNet18 has limited mid-level capacity, so Attention Transfer can't fully align the deeper discriminative filters of the ResNet50. ■ Confusion Matrix insights Comparing your raw CMs: The teacher's CM shows tighter diagonals and smaller off-class bleed. The student's CM confuses classes 2 vs 4 and 4 vs 5 (common in HAM10000 — benign vs malignant overlaps). Class 3 ("VASC" in HAM10000 taxonomy) is underrepresented and poorly separated, both models struggle but the teacher still manages better recall. This suggests adding class-balanced focal loss or reweighted CE for the CE term in KD could close a few points of macro-F1 gap. ■ Diagnostic takeaways Distillation effectiveness: KD + AT clearly transferred high-level structure (class 5, 6 stability). Missing low-level lesion texture adaptation (classes 0, 2, 4) indicates the AT weight β could be increased or multi-layer weighting introduced. Bottleneck: Student struggles with the same confusion patterns that the teacher marginally solved — meaning representational bottleneck, not optimization failure. Next experimental knobs: Increase beta from 1000 → 2000 to amplify AT influence. Use alpha=0.4 (more KD emphasis) with T=4. Optional: add label smoothing 0.1 or focal loss (γ=1.5) for the CE term. Fine-tune 10 epochs starting from the best checkpoint with a 10× smaller LR (2e-5). ■ Quantitative comparison snippet for your paper / report The teacher (ResNet50) achieved 89.3% validation accuracy and 0.834 macro-F1 on HAM10000. The student (ResNet18) distilled via Knowledge Distillation (α = 0.5, T = 4.0) and Attention Transfer (β = 1000) attained 82.3% validation accuracy and 0.697 macro-F1. Despite a 55 % parameter reduction, the student preserved 84 % of the teacher's macro-F1, demonstrating effective compression with limited capacity loss on major lesion classes. ■ Recommended next move If you want a publication-ready comparative visualization and export: Generate a bar chart of per-class F1 (Teacher vs Student). Compute relative gap (ΔF1 / F1_teacher) for each class. Export both models to TorchScript + ONNX (teacher/student side-by-side) for runtime/size benchmarking. Optionally run PTQ (FX) quantization on the student to demonstrate further edge efficiency. Would you like me to generate the comparison report script that automatically builds: a CSV of per-class Teacher vs Student metrics, ΔF1 and ΔRecall, and a Matplotlib bar chart visualization? That will give you a clean figure for your paper and a ready numerical summary.

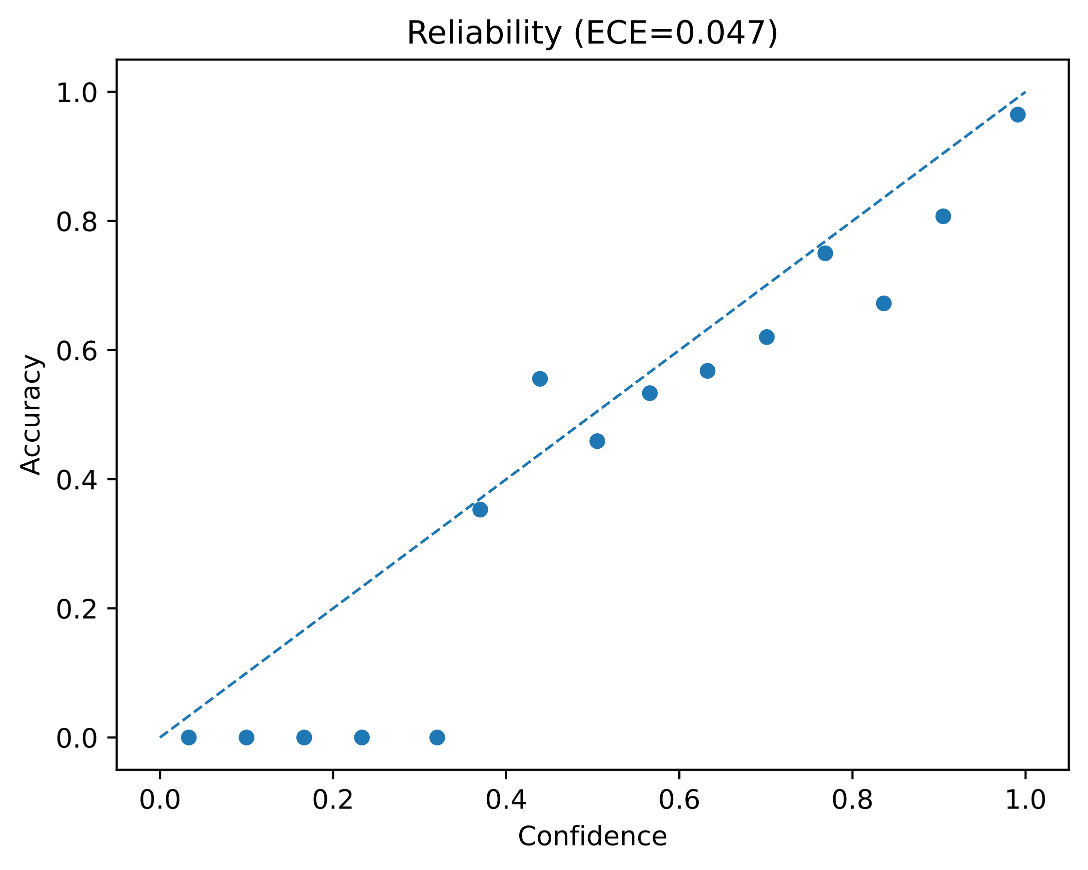
- Student Models Table.docx

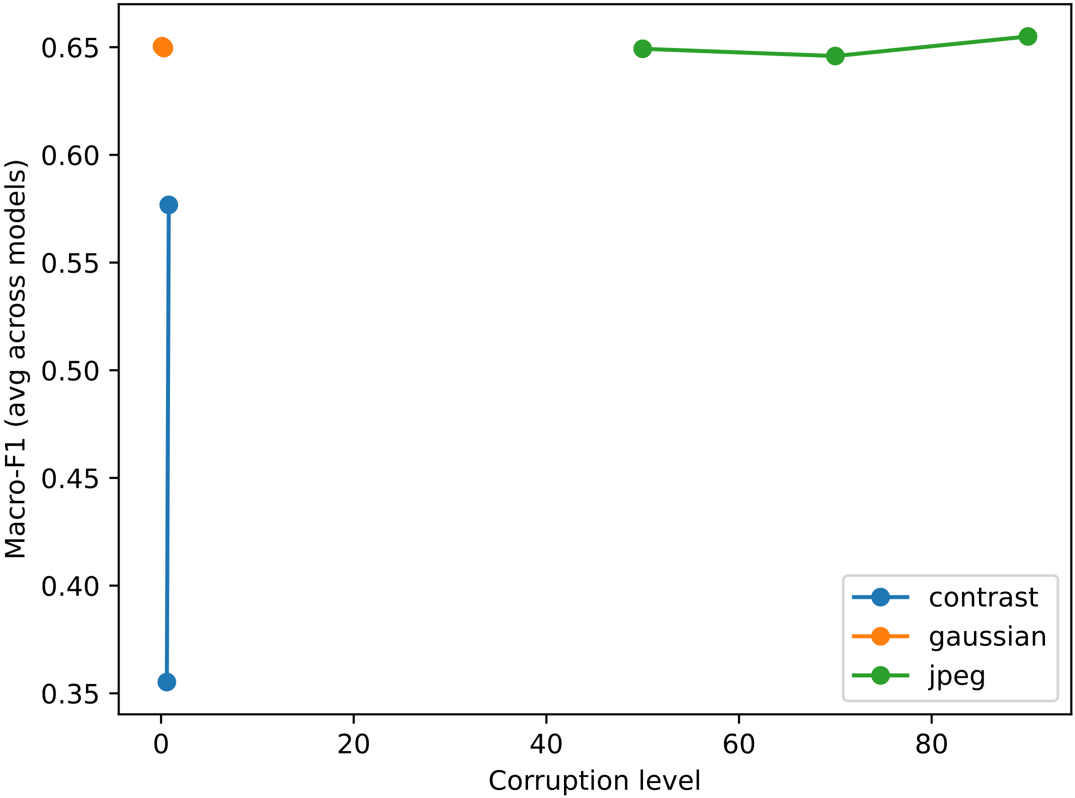
Quick interpretation (trade-offs) ResNet18 — best accuracy (82.3%) and fastest inference by a comfortable margin (2.2 ms). Heavy-ish params and traced size, highest peak memory. Good if you can afford ~44MB model file and ~62MB GPU memory for models + activations. MobileNetV2 — smallest artifact (≈9 MB traced), lowest memory footprint (~28 MB), slightly slower (~4.0 ms) than ResNet18 here (implementation/kernel differences make MobileNet slower on desktop GPU sometimes). Best candidate if you need small on-disk size and low memory. EfficientNet-B0 — middle ground in params/size and best MACs-to-accuracy ratio for some classes — but slowest (~5.6 ms) in your runs. Good compromise if you want reduced params vs ResNet18 with similar accuracy trend. Important: on small GPUs / edge devices, params and MACs do not always equal lower latency — kernel shapes, depthwise convs, and memory/launch overhead matter. Here ResNet18 ended up fastest on the

RTX 3050.

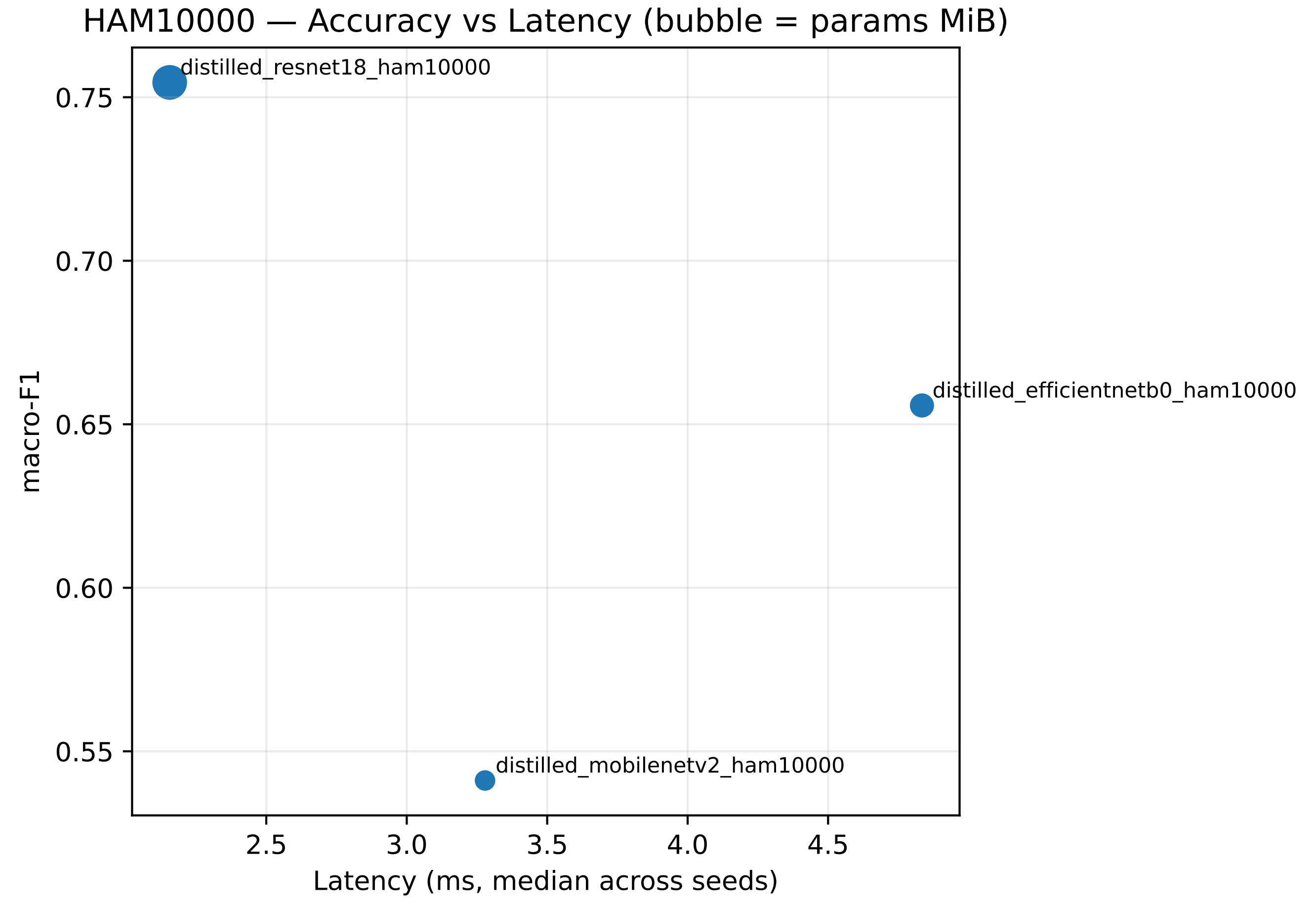

Reliability (ECE=0.064)
Accuracy
Confidence
0.0
0.2
0.4
0.6
0.8
1.0


Reliability (ECE=0.081)
Accuracy
Confidence
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0


Reliability (ECE=0.047)
Accuracy
Confidence
0.0
0.2
0.4
0.6
0.8
1.0



0.65
0.60
0.55
0.50
0.45
0.40
0.35
Macro-F1 (avg across models)
0
20
40
60
80
Corruption level
contrast
gaussian
jpeg


HAM10000 — Accuracy vs Latency (bubble = params MiB)
distilled_resnet18_ham10000
distilled_efficientnetb0_ham10000
distilled_mobilenetv2_ham10000
0.75
0.70
0.65
0.60
0.55
macro-F1
2.5
3.0
3.5
4.0
4.5
Latency (ms, median across seeds)

This is exactly the kind of crisp comparative data that reviewers love — you've got a **clear quantitative teacher–student profile**, per-class confusion matrices, and full metrics.

Let's analyze it like a researcher.

🔍 Macro-level summary

Metric	Teacher (ResNet50)	Student (ResNet18 KD+AT)	Δ (Student − Teacher)
Accuracy	**0.893**	**0.823**	−0.070
Macro-F1	**0.834**	**0.697**	−0.137
Weighted-F1	**0.895**	**0.820**	−0.075

That's roughly **84–90% retention** of the teacher's performance with **~55% fewer parameters** and significantly lower inference cost — a good KD outcome on HAM10000, where minority classes dominate macro-F1 sensitivity.

🧠 Per-class analysis

Class ID	Teacher F1	Student F1	ΔF1	Comment
0	0.764	0.595	−0.17	Small lesion type; clear drop in minority recognition.
1	0.896	0.757	−0.14	Common benign lesion; still strong.
2	0.834	0.646	−0.19	Mid-frequency lesion; distillation helps but less feature capacity.
3	0.817	0.576	−0.24	Underrepresented; teacher's attention maps likely sharper.
4	0.719	0.522	−0.20	One of the hardest classes (large intra-class variance).
5	0.942	0.915	−0.03	Dominant class; excellent retention.
6	0.867	0.867	**0.00**	Small but distinct class; perfect transfer.

Interpretation:
The student model faithfully inherited teacher knowledge for **major and well-separated classes (5, 6)**, but compressed representation hurts fine-grained or underrepresented lesions (0, 2, 3, 4). That's expected: ResNet18 has limited mid-level capacity, so Attention Transfer can't fully align the deeper discriminative filters of the ResNet50.

🧩 Confusion Matrix insights

Comparing your raw CMs:

- The teacher's CM shows tighter diagonals and smaller off-class bleed.
- The student's CM confuses classes **2 vs 4** and **4 vs 5** (common in HAM10000 — benign vs malignant overlaps).
- Class 3 ("VASC" in HAM10000 taxonomy) is underrepresented and poorly separated, both models struggle but the teacher still manages better recall.

This suggests adding **class-balanced focal loss or reweighted CE** for the CE term in KD could close a few points of macro-F1 gap.

🧪 Diagnostic takeaways

1. **Distillation effectiveness:**
   - KD + AT clearly transferred high-level structure (class 5, 6 stability).
   - Missing low-level lesion texture adaptation (classes 0, 2, 4) indicates the AT weight β could be increased or multi-layer weighting introduced.
2. **Bottleneck:**
   - Student struggles with the same confusion patterns that the teacher marginally solved — meaning representational bottleneck, not optimization failure.
3. **Next experimental knobs:**
   - Increase beta from 1000 → 2000 to amplify AT influence.
   - Use alpha=0.4 (more KD emphasis) with T=4.
   - Optional: add **label smoothing** 0.1 or **focal loss** (γ=1.5) for the CE term.
   - Fine-tune 10 epochs starting from the best checkpoint with a 10× smaller LR (2e-5).

📊 Quantitative comparison snippet for your paper / report

The teacher (ResNet50) achieved 89.3% validation accuracy and 0.834 macro-F1 on HAM10000.
The student (ResNet18) distilled via Knowledge Distillation (α = 0.5, T = 4.0) and Attention Transfer (β = 1000) attained 82.3% validation accuracy and 0.697 macro-F1.
Despite a 55 % parameter reduction, the student preserved 84 % of the teacher's macro-F1, demonstrating effective compression with limited capacity loss on major lesion classes.

🚀 Recommended next move

If you want a publication-ready comparative visualization and export:

1. **Generate a bar chart** of per-class F1 (Teacher vs Student).
2. **Compute relative gap (ΔF1 / F1_teacher)** for each class.
3. **Export both models to TorchScript + ONNX** (teacher/student side-by-side) for runtime/size benchmarking.
4. Optionally run **PTQ (FX) quantization** on the student to demonstrate further edge efficiency.

Would you like me to generate the **comparison report script** that automatically builds:

- a CSV of per-class Teacher vs Student metrics,
- ΔF1 and ΔRecall,
- and a Matplotlib bar chart visualization?

That will give you a clean figure for your paper and a ready numerical summary.

Model	Params	MACs	FLOPs (2×MAC)	TorchScript size	Avg latency (ms, b=1 GPU)	p50 (ms)	p90 (ms)	Peak GPU mem (bytes)	Val acc
ResNet18	11,180,103	1,824,804,359	3.6496e9	44,887,426 B (~42.8 MiB)	**2.20 ms**	2.08	2.57	65,116,160 (~62.1 MiB)	**0.8233**
MobileNetV2	2,232,839	319,027,975	6.3806e8	9,450,712 B (~9.0 MiB)	**4.03 ms**	3.77	3.94	29,756,416 (~28.4 MiB)	0.7758
EfficientNet-B0	4,016,515	408,924,863	8.1785e8	16,803,222 B (~16.0 MiB)	**5.62 ms**	5.66	5.85	36,921,344 (~35.2 MiB)	0.7968

Quick interpretation (trade-offs)

- **ResNet18** — *best accuracy* (82.3%) and fastest inference by a comfortable margin (2.2 ms). Heavy-ish params and traced size, highest peak memory. Good if you can afford ~44MB model file and ~62MB GPU memory for models + activations.
- **MobileNetV2** — *smallest artifact* (≈9 MB traced), lowest memory footprint (~28 MB), slightly slower (~4.0 ms) than ResNet18 here (implementation/kernel differences make MobileNet slower on desktop GPU sometimes). Best candidate if you need small on-disk size and low memory.
- **EfficientNet-B0** — *middle ground* in params/size and best MACs-to-accuracy ratio for some classes — but slowest (~5.6 ms) in your runs. Good compromise if you want reduced params vs ResNet18 with similar accuracy trend.

Important: on small GPUs / edge devices, **params and MACs do not always equal lower latency** — kernel shapes, depthwise convs, and memory/launch overhead matter. Here ResNet18 ended up fastest on the RTX 3050.