Project G5: Melanoma Detection using Knowledge Distillation for Mobile Phones

Milestone-II progress evaluation

- The focus on HAM10000 with ∼10,000 dermoscopic images and an ≈ 11% melanoma prevalence is clearly stated and clinically relevant.
- **Strong literature context for KD + edge deployment**: The survey of Islam, Kabir, Nazari, Polino, Touvron, and Gou is well chosen; you explicitly position your work at the intersection of melanoma detection, knowledge distillation, and model compression / quantization.
- **Baseline sanity checks are a good start but not yet image-centric**: Your first experiments with GBM, logistic regression, and random forests using handcrafted pixel features and metadata yield Acc ≈ 0.89–0.90 and ROC–AUC ≈ 0.86–0.90, but you correctly flag that with only ∼ 10% melanoma, these models may not truly be "seeing" melanoma patterns in the images. This is more of a label-distribution / metadata sanity check than a real vision baseline.
- **Teacher model is partially implemented, but plan and implementation do not fully match**: The experiment plan promises a ResNet50 teacher, but the preliminary results section reports a ResNet18 with 11,177,538 total parameters, only 1,026 trainable, and validation accuracy = 84.4% at epoch 11. This is a useful first step, but:
  - There is no ROC–AUC, PR–AUC, or sensitivity-at-high-recall reported yet for the teacher.
  - It is unclear whether ResNet18 will remain the final teacher, or whether you will still compare ResNet18 vs ResNet34 vs ResNet50 as originally envisioned.
- **KD and mobile deployment are still only planned, not executed**: The KD loss, temperature T, mixing weight α, and quantization settings are described in the experiment plan, and the evaluation plan specifies ROC–AUC, PR–AUC, and metrics at 95% sensitivity, but there are no student-model or quantized-model results yet.
- **Search space is still too large for the actual progress to date**: The proposed grid over loss types (BCE, weighted BCE, focal), pooling variants, T, α, learning rate, batch size, dropout, and bitwidths is huge relative to what is currently implemented. Without aggressive pruning, this will take a lot of training time even on Rivanna GPUs.
- **Minor but important inconsistencies**: The methodology mentions both a standard 80/20 train–holdout split and a 70/15/15 train–test–holdout split. These must be reconciled into one clearly defined protocol.

Core fixes before adding any new KD / student models

- **Standardize and document the data split protocol**:
  - Pick a single strategy, e.g. stratified 70/15/15 (train/val/holdout) on the melanoma vs non-melanoma label.
  - Ensure that images from the same lesion ID never appear in both train and holdout (HAM10000 has multiple images per lesion). This is critical to avoid overly optimistic performance due to near-duplicate images leaking across splits.
  - Clearly state: total N, number of melanoma vs non-melanoma samples in each split, and random seed used.
- **Move from metadata baselines to a strong *image-based* teacher baseline**:
  - Treat the GBM / RF / logistic models on handcrafted pixel + metadata as sanity checks, not central results. They are useful to confirm labels and class imbalance, but they will not convince reviewers that you can recognize melanoma in images.

  - Train a single strong teacher CNN (e.g. ResNet18 or ResNet34) end-to-end on dermoscopic images with standard augmentations (random flips, rotations, mild color jitter, random crops).
  - Use a class-balanced loss (weighted BCE or focal loss with $\gamma \approx 2$) so that the teacher achieves high sensitivity on melanoma, not just high overall accuracy.

- **Report full teacher performance *before* distillation**:
  - For the teacher, report at least:

    $$\text{ROC–AUC}, \quad \text{PR–AUC}, \quad \text{Sensitivity at fixed specificity}, \quad \text{Specificity at 95\% sensitivity}.$$

  - Add calibration metrics (e.g. Expected Calibration Error, ECE) and a reliability diagram on the holdout set. KD from an uncalibrated teacher is often suboptimal.

- **Clarify teacher architecture choice and freezing strategy**:
  - Decide whether your final teacher is ResNet18, 34, or 50. If you stay with ResNet18 for computational reasons, say so explicitly and motivate the choice.
  - Make sure the freezing / fine-tuning setup is correct (e.g. backbone frozen vs partially unfrozen). The extremely small number of trainable parameters ($\sim 10^3$) suggests you may only be training the very last layer; this is fine as a starting point but may underfit the dataset.

- **Tighten the classical baselines**:
  - For the GBM / RF / logistic baselines, consider evaluating them *only* on simple features like metadata and perhaps a global color histogram. Keep their role limited to "tabular vs image" comparison.
  - Clearly state that these baselines are class-imbalance-aware (e.g. class weights or resampling) and report melanoma-specific recall and precision.

Knowledge distillation design

- **Write down the exact KD loss you will implement**. For instance a binary classification setting, a standard KD objective is:

  $$\mathcal{L}_{\text{KD}} = \alpha T^2 \cdot \text{KL}\big(p_T^{(\text{teacher})} \,\|\, p_T^{(\text{student})}\big) + (1-\alpha) \cdot \mathcal{L}_{\text{BCE}}\big(y, p^{(\text{student})}\big),$$

  where $p_T = \sigma(z/T)$ are temperature-scaled logits, and $\mathcal{L}_{\text{BCE}}$ is the usual binary cross-entropy between ground-truth labels y and student predictions.

- **Aggressively shrink the KD search space**:
  - Focus on a very small set of KD hyperparameters:

    $$T \in \{1, 2\}, \qquad \alpha \in \{0.5, 0.9\}.$$

    This already gives you four KD settings and matches common practice from the literature (e.g. Islam et al. found simple settings effective on HAM10000).
  - Fix the loss type (e.g. weighted BCE) and student pooling (e.g. global max pooling) for the main experiments, and only change these in 1–2 targeted ablations.
  - Defer large sweeps over learning rate, batch size, and dropout. If you tune them at all, do a simple two-point sweep (e.g. $\eta \in \{10^{-3}, 10^{-4}\}$) while holding everything else fixed.

- **Teacher–student architecture choices**:
  - Fix a single student architecture such as MobileNetV3-Small (or even a smaller variant) that cleanly fits your target model-size budgets (e.g. ≤ 25 MB for mobile and ≤ 2 MB for edge).
  - Keep the teacher fixed when you study T, α, and losses; do not simultaneously change teacher and student when analyzing KD behavior, or it will be hard to interpret results.

- **Quantization as a separate, clearly defined stage**:
  - Once you have a well-performing student, treat quantization as a second stage: e.g. baseline fp32 student → post-training int8 quantization. Also measure the drop in ROC–AUC and the change in ECE:

$$\Delta\text{AUC} = \text{AUC}_{\text{fp32}} - \text{AUC}_{\text{int8}}, \qquad \Delta\text{ECE} = \text{ECE}_{\text{int8}} - \text{ECE}_{\text{fp32}}.$$

  - Only if post-training quantization degrades performance substantially (e.g. $\Delta\text{AUC} > 0.02$ or $\Delta\text{ECE} > 0.05$), consider a single quantization-aware-training (QAT) experiment.

Evaluation design and mobile / clinical flavor

- **Operational metrics at fixed sensitivity**: You already plan to choose a threshold at 95% sensitivity and report specificity, PPV, and NPV. Make this central to the story:
  - For each model (teacher, student, quantized student), report:

$$\text{ROC–AUC},\ \text{PR–AUC},\ \text{Sensitivity},\ \text{Specificity},\ \text{PPV},\ \text{NPV}$$

    at the chosen high-sensitivity operating point.
  - Compare these metrics teacher vs student vs quantized student to illustrate the trade-off between accuracy, calibration, and deployability.
- **Calibration and uncertainty**:
  - Explicitly compute Expected Calibration Error (ECE) on the holdout set for all three models.
  - Add reliability diagrams (predicted probability vs empirical frequency) for teacher and student.
  - If you apply temperature scaling to the teacher, report pre- and post-scaling ECE and ROC–AUC to show that the teacher becomes a better "soft-label provider".
- **Mobile and edge deployment metrics**:
  - For teacher and student (fp32 and int8), report:

$$\text{Parameter count},\ \text{model size (MB)},\ \text{FLOPs},\ \text{CPU latency (ms / image)}.$$

  - Given the remaining time, you cannot test on a real phone, so approximate mobile inference by measuring latency on a single CPU core and clearly documenting the hardware (e.g. "M3 CPU core, batch size 1").
  - Highlight whether the student (and quantized student) meet your target budgets (e.g. ≤ 25 MB, ≤ 100 ms / image at batch size 1).
- **Clinical discussion**: Reserve a short subsection to discuss:
  - The risk of false negatives (missed melanoma) vs false positives (unnecessary referrals).
  - How an app based on your model might be used in practice (e.g. as a pre-screening tool that recommends dermatologist visits rather than giving a definitive diagnosis).
  - Limitations: dermoscopy images vs real smartphone photos, dataset bias (skin tones, lesion locations), and potential strategies to mitigate these in future work.

Code and repository hygiene

- **Repository URL and accessibility**:
  - Add a top-level `README.md` that includes: dataset download instructions, environment setup, and exact commands (or notebooks) to reproduce teacher training, KD experiments, and quantization.
- **Modularize the code**:
  - Separate components into modules such as:
    - `data.py` (HAM10000 loading, preprocessing, stratified splits, augmentations)
    - `models.py` (teacher, student architectures)
    - `kd_loss.py` (implementation of $\mathcal{L}_{\text{KD}}$)
    - `train_teacher.py`, `train_student.py`
    - `eval.py` (metrics, calibration, latency, size)
  - Keep Jupyter notebooks primarily for exploratory analysis and plotting.
- **Reproducibility and logging**:
  - Fix random seeds and document them.
  - Use W&B or a simple CSV / JSON logger to record configuration (teacher/student architecture, T, α, loss type, learning rate), metrics, and timestamps for each run.
  - Save the best model checkpoints (based on validation AUC or sensitivity at high recall) and record which checkpoint produced the final holdout results.

Overall guidance for the final report

- **Tell a focused story rather than exploring everything**:
  - A compelling final project would look like:
    1. Strong, calibrated teacher CNN for melanoma vs non-melanoma on HAM10000.
    2. Carefully distilled MobileNetV3-Small student with comparable sensitivity and calibration.
    3. Quantized student that fits edge-device budgets with minimal performance loss.

    plus a small number of clean ablations on T and α.
- **Align experiments with your stated gaps**:
  - You emphasized that prior KD work often neglects sensitivity at a fixed operating point and calibration. Make sure your *results* actually address those gaps: highlight high-sensitivity operating points, ECE, and reliability diagrams in the main figures.
- **Make contributions explicit in the conclusion**:
  - For example: "We show that a distilled MobileNetV3-Small student can match the ROC–AUC and high-sensitivity performance of a larger ResNet teacher while reducing model size by $\times k$ and latency by $\times m$, with only a small change in ECE after int8 quantization."
  - This type of crisp quantitative statement will make the final report read like a proper machine learning paper rather than a course project.