Project G13: Toxic Plant Classification

Milestone-II progress evaluation

- **Problem, dataset, and research question are well framed**: You clearly motivate the real-world risk of misidentifying poisonous plants and pose a research question centered on how architecture and hyperparameter choices affect benign vs toxic classification across diverse species and backgrounds. The Kaggle dataset description is precise: ~10,000 images, 10 classes (5 toxic, 5 visually similar non-toxic), with a stratified 69/15/15 train/validation/test split by species and toxicity.

- **Baseline and deep models are well aligned with your RQ**: You implemented both a feature-based Random Forest baseline and an EfficientNet-B0 CNN (with and without CBAM, with and without ImageNet pretraining). This matches your stated goal of comparing traditional ML vs modern CNNs for toxic plant classification.

- **Non-DL baseline is meaningful and justifies CNN**: The tuned Random Forest (GLCM + shape features) reaches $\approx 60\%$ accuracy and similar F1 for toxic vs non-toxic classes, which is only modestly above chance. This strongly justifies moving to CNNs and makes the later EfficientNet-B0 results more compelling.

- **EfficientNet-B0 + CBAM + pretraining is promising but clearly overfits**: Your best model (EfficientNet-B0 with CBAM and pretrained weights) achieves $\approx 99\%$ training accuracy, $\approx 83\%$ validation accuracy, and $\approx 82\%$ test accuracy. This is a large improvement over the Random Forest and other CNN variants, and directly supports the value of both pretraining and attention. At the same time, the gap between training and validation accuracy indicates substantial overfitting that you should address explicitly.

- **Current evaluation is accuracy-/precision-heavy and binary-only**: For both RF and CNN, you primarily report overall accuracy and, for some models, precision. Given the safety-critical nature of toxic-plant detection and the balanced dataset, you should still emphasise per-class recall and F1, especially for the toxic class. In addition, your experiments are currently restricted to the binary toxicity label, while the dataset also supports a 10-class species task that you mention only as a possible extension.

- **Ablations are already started but need clearer framing**: You effectively ran four model variants (RF vs tuned RF, EfficientNet-B0 random vs pretrained, with vs without CBAM), but they are currently just listed as "variants" rather than positioned as clean ablation factors (pretraining on/off, CBAM on/off, augmentation strength). Making this explicit in tables and text will make your final paper read much more like a proper ML study.

Core fixes before adding new architectures

- **Prioritise toxic-class recall and F1, not only accuracy**:
  - For each model (RF, tuned RF, all EfficientNet-B0 variants), report per-class precision, recall, and F1 for the binary task:

$$F1_{\text{toxic}} = \frac{2\,\mathrm{TP}_{\text{toxic}}}{2\,\mathrm{TP}_{\text{toxic}} + \mathrm{FP}_{\text{toxic}} + \mathrm{FN}_{\text{toxic}}}, \quad F1_{\text{non-toxic}} = \frac{2\,\mathrm{TP}_{\text{non}}}{2\,\mathrm{TP}_{\text{non}} + \mathrm{FP}_{\text{non}} + \mathrm{FN}_{\text{non}}}.$$

  - Highlight explicitly the false-negative rate for the toxic class (toxic plants misclassified as safe). For a real-world app, false negatives are much more dangerous than false positives.
  - Add ROC curves and AUC for the best model(s), and consider choosing an operating threshold that targets *high toxic recall*, then report corresponding precision.

- **Make the primary task explicit**:
  - State clearly that the main supervised task for this project is *binary toxicity classification*, with 10-class species recognition as an optional extension if time permits. This will anchor your evaluation choices and keep the scope realistic.

- If you later add a species head, treat it as a separate section (or multi-task extension) rather than letting it blur the main toxicity results.

- **Address overfitting in the CBAM + pretrained model**:
  - A training accuracy of $\sim 99\%$ vs validation $\sim 83\%$ suggests your model is memorising training images. Before scaling up to larger EfficientNet variants (B1–B7), work on:
    - Stronger data augmentation (see ablation suggestions below).
    - Regularisation: add dropout in the classifier head (e.g., $p = 0.3$–0.5) and ensure you have non-zero weight decay.
    - Early stopping based on validation loss / accuracy rather than training for a fixed 10 epochs. Monitor whether validation performance plateaus earlier.
  - Once you reduce overfitting with B0, you can more confidently justify exploring B1/B2. Jumping to B7 without controlling overfitting will likely just make things worse.

- **Clarify feature extraction details for the RF baseline**:
  - Document exactly how you constructed RGB histograms (number of bins per channel, normalisation), the GLCM parameters (offsets, angles, distance, and which statistics: contrast, energy, homogeneity), and how circularity and aspect ratio were computed.
  - These details are important for reproducibility and will make the RF baseline more credible in the final paper.

Ablation and experiment design

- **Turn your four CNN variants into a clean 2×2 ablation**:
  - You already effectively have:

$$\text{Pretraining} \in \{\text{off},\ \text{on}\}, \quad \text{CBAM} \in \{\text{off},\ \text{on}\}$$

    with EfficientNet-B0 and light augmentation.
  - Present a table where each cell (off/on) reports validation and test accuracy, precision, recall, and $F1_{\text{toxic}}$. This will clearly show the marginal effect of pretraining and of CBAM.

- **Add a focused augmentation ablation**:
  - Define two augmentation regimes:

    Light: current setup (e.g., random horizontal flip, small rotations).

    Heavy: light + color jitter (brightness/contrast/saturation), random resized crops, and possibly Cutout or MixUp to force robustness to background and occlusions.
  - Train the best-performing EfficientNet-B0 variant (pretrained + CBAM) under both regimes with the same hyperparameters, and compare:

$$\text{Acc}, \quad F1_{\text{toxic}}, \quad F1_{\text{non-toxic}}$$

    on the *test* set. A gain of 5–10% in test performance under heavy augmentation would be a strong, publishable result given the iNaturalist-style background diversity.

- **Pretraining vs random initialisation ablation**:
  - For the CBAM configuration, explicitly compare:

$$\text{EffB0+CBAM (random init)} \quad \text{vs} \quad \text{EffB0+CBAM (ImageNet pretrained)}$$

    with the same augmentation and training schedule. Quantify the lift in test accuracy and $F1_{\text{toxic}}$ from pretraining (likely +10–15%).

  - This directly addresses your RQ about architecture and hyperparameter choices, and shows the value of transfer learning in this plant domain.

- **If time permits, a small "B0 vs B2" comparison is enough**:
  - Rather than sweeping EfficientNet-B1–B7, pick a single slightly larger model (e.g., B2) and compare against B0 under your best augmentation + regularisation regime.
  - If B2 gives only marginal gains or overfits, you have a strong argument that B0 is sufficient for this dataset size and task.

Evaluation design and safety / usability

- **Design evaluation around toxic safety, not just average performance**:
  - For the final model, compute and report:

$$\text{ROC curve, AUC,}\quad \text{Precision–Recall curve for toxic class.}$$

  - Choose a decision threshold that achieves high recall for toxic plants (e.g., $R_{\text{toxic}} \geq 0.95$ on the validation set), then report test-time precision and F1 at that threshold. This allows you to say something like: "We detect 95% of toxic plants at the cost of X% false alarms."

- **Use confusion matrices to interpret specific plant pairs**:
  - Even though your primary task is binary, construct confusion matrices stratified by species (e.g., which non-toxic species most often get mislabelled as toxic, and vice versa).
  - Specifically discuss pairs such as poison ivy vs Virginia creeper or western vs eastern variants. This kind of analysis will make the paper much richer and more botanically grounded.

- **Guard against background overfitting**:
  - Use Grad-CAM or similar saliency methods on the best model to visualise which image regions drive toxic predictions. Check whether the model focuses on leaves / leaflets and not on background textures.
  - If you find strong background dependence, mention this as a limitation and connect it to your augmentation and regularisation plans.

- **Add a short "not for unsupervised medical use" disclaimer**:
  - In the introduction or conclusion, clarify that this is a proof-of-concept model trained on curated images and is not a replacement for expert identification in critical situations. This makes the project more responsible and realistic.

Code and repository hygiene

- **Ensure the GitHub repo is reproducible**:
  - Include a clear `README.md` describing dataset download, environment setup, and commands / notebooks to reproduce:
    1. Random Forest baseline,
    2. EfficientNet-B0 (all four variants),
    3. Ablation experiments (augmentation, pretraining, CBAM).
  - Add a `requirements.txt` or `environment.yml` pinning versions of `torch`, `torchvision`, `scikit-learn`, etc.

- **Modularise the pipeline**:
  - If not already done, separate code into modules such as:

    - `data.py` (dataset class, transforms, stratified splits),
    - `features_rf.py` (histograms, GLCM, shape for RF),
    - `models.py` (EfficientNet-B0, CBAM wrapper),
    - `train_rf.py`, `train_cnn.py`,
    - `eval.py` (metrics, confusion matrices, ROC/AUC).
  - Keep Jupyter notebooks mostly for visualisations (loss curves, Grad-CAM, example predictions).

- **Experiment logging**:
  - Use W&B, MLflow, or simple CSV logs to record, for each run: model variant, augmentation regime, seed, hyperparameters (learning rate, epochs, weight decay, dropout), and metrics on train/val/test.
  - This will make it easy to construct final tables and plots and to argue convincingly about which design choices matter.

Overall guidance for the final report

- **Tell a focused, safety-aware story**: For example,

  > "We show that an EfficientNet-B0 CNN with CBAM attention and ImageNet pretraining substantially outperforms a traditional Random Forest baseline for toxic plant identification on a balanced Kaggle dataset. Through ablations on pretraining, attention, and augmentation, we quantify which design choices most improve toxic-class recall while controlling overfitting."

- **Make contributions explicit and quantitative**:
  - A concise conclusion might say something like: "Compared to a tuned Random Forest baseline (60% accuracy), our best CNN achieves 82% accuracy and improves toxic-class F1 by X points, while strong data augmentation and attention reduce errors on challenging toxic/non-toxic lookalike pairs."

- **Keep complexity manageable**: Depth over breadth will make this feel like a small conference paper. A well-executed comparison of RF vs EfficientNet-B0 variants (with clean ablations and safety-aware metrics) will be much stronger than a rushed exploration of many larger architectures without careful analysis.