Flower Species Classification using CNN (Oxford 102)

ResNet-50 transfer learning identifies 102 flower species from the Oxford 102 Flowers dataset, with a web demo ranking top-5 species.

Title

Flower Species Classification using CNN (Oxford 102) — fine-grained visual recognition via transfer learning.

Introduction / Background

Distinguishing 102 flower species — many differing only in petal texture or stamen shape — is fine-grained classification, a genuinely hard vision problem. The Oxford 102 Flowers dataset (8,189 photographs, 102 categories, some with only ~40 images) is the classic benchmark: too small for training deep networks from scratch, too fine-grained for shallow ones. The rigorous answer is transfer learning done in two phases: a frozen-backbone linear probe, then careful fine-tuning.

Problem Statement

Naive fine-tuning of a deep network on 8k images overfits or destroys pretrained features. A disciplined two-phase schedule with layer-wise learning rates, augmentation and top-k evaluation is needed — plus a demo that makes the classifier tangible, like a plant-identification app.

Objectives

- Fine-tune ResNet-50 to classify 102 flower species.
- Implement and document the linear-probe-then-fine-tune schedule.
- Evaluate with top-1/top-5 accuracy, macro F1 and confusable-species analysis.
- Build a web demo ranking top-5 species for bloom photos.
- Document training decisions for viva readiness.

Proposed Methodology

Load Oxford 102 Flowers with official splits (train+validation combined for fitting). Resize to 224×224, ImageNet normalization; augmentation with random crops, flips and color jitter. Phase 1: train only the new head on a frozen ResNet-50 backbone. Phase 2: unfreeze top residual blocks, continue at 10× lower learning rate with early stopping. Evaluate once on the held-out test set.

System Architecture / Working

Training pipeline (PyTorch/Jupyter): data → preprocessing → augmentation → phase-1 probe → phase-2 fine-tune → evaluation → error gallery. Demo (web app): bloom photo → identical preprocessing → saved weights → softmax → ranked top-5 species with confidence bars and Latin names.

Hardware / Software Requirements

- Software: Python 3, PyTorch/torchvision, PIL, NumPy, Matplotlib, scikit-learn, Jupyter; web demo in HTML/CSS/JavaScript (runs offline).
- Hardware: any laptop for the demo; training on a free Colab GPU (approx. 1–2 hrs expected). Dataset: Oxford 102 Flowers, VGG, University of Oxford.

Expected Outcomes

A 102-class classifier with design-target top-1 accuracy ≈94% and top-5 ≈99% (actual figures documented after the training run); training curves; misclassified-sample gallery; a working species-ranking demo.

Applications

- Plant-identification style demo apps.
- Teaching transfer learning and fine-grained evaluation.
- Gardening and nursery assistance prototypes.
- Viva demonstration of a rigorous deep-learning pipeline.

Future Scope

- Mobile deployment for in-the-field identification.
- Extension to leaves, bark and full-plant images.
- Few-shot learning for rare species.
- Geolocation priors to re-rank predictions.