Vision Project Proposal

By: Saige Young

Abstract:

This project investigates the application of deep learning and transfer learning to classify weather conditions (Cloudy, Rain, Shine, Sunrise) using image data. We evaluate and compare multiple pre-trained convolutional neural networks (CNNs) to determine which model performs best on the task. The goal is to build an accurate, generalizable image classification system for real-world weather prediction.

Introduction:

Weather classification using image data has wide-reaching implications, especially for agriculture, transportation, and disaster preparedness. By leveraging deep learning and transfer learning with models like MobileNetV2, ResNet50, and DenseNet121, we aim to develop a robust classifier for weather imagery. This presentation outlines the modeling approach, evaluation metrics, dataset, and experimental workflow.

Motivation:

Computer vision applications in meteorology are growing due to increased availability of weather image datasets and advancements in transfer learning. Accurately classifying weather from images is useful for automation and early warning systems. This project explores which CNN architecture generalizes best and how data augmentation affects performance. Previous works in this space have mostly used traditional CNNs or simpler datasets; our study aims to apply modern architectures and visualize performance through metrics like ROC curves.

Problem Formulation

- **Is this a new problem?**

  This is an extension of prior weather classification tasks using advanced transfer learning models.

- **Datasets:**

  The dataset includes labeled images categorized into 4 weather classes: Cloudy, Rain, Shine, Sunrise. The data was split into training and validation sets, and preprocessing included resizing, normalization, and RGB conversion.

  - **Type/size:** JPEG/PNG images, resized to 180x180 or 224x224, with ~100–300 samples per class.
  - **Resources/tools:** TensorFlow, Keras, Matplotlib, scikit-learn. Pre-trained models from keras.applications.
  - **Processing needed:** Conversion from grayscale to RGB, augmentation (rotation, zoom, flip).

- **Goal/task:**

  Image classification. The task maps to multi-class classification using CNN-based transfer learning.

- **Libraries used:**

  TensorFlow/Keras, scikit-learn for metrics, Matplotlib for visualization.

- **Metrics:**
  Accuracy, F1-score, ROC-AUC for each class. Performance is estimated based on literature benchmarks (typically 85–95% accuracy).

- **Train/validation/testing approach:**
  Use 80% of the dataset for training and 20% for validation. ROC curves are used to compare the models.

- **Workplan**
  - **Week 1:** Preprocess dataset, load and visualize samples, define baseline CNN.
  - **Week 2:** Train MobileNetV2 and evaluate. Add data augmentation.
  - **Week 3:** Train ResNet50 and DenseNet121. Generate classification reports and ROC curves.
  - **Week 4:** Compare results across all three models. Finalize visualizations and write report.