Deep Learning - Exercise 1

https://github.com/NaamaAvn/deep-learning-ex1/blob/main/project.ipynb

Naama Avni 208523456

Problem Definition

Objective: To build an image classifier that can automatically distinguish between winter and summer scenes in photographs.
I chose this problem because it represents a complex visual understanding task that requires the model to recognize detailed features beyond simple color patterns. I also selected it because I enjoy working with nature imagery.

Classification Task: Binary image classification

- Input: Digital images (photographs)
- Output: Class label (winter or summer)

Classes:

1. Winter: Images containing winter scenes characterized by:
   - Snow and ice
   - Bare trees
   - Winter clothing
   - Cold-weather landscapes
   - Winter activities

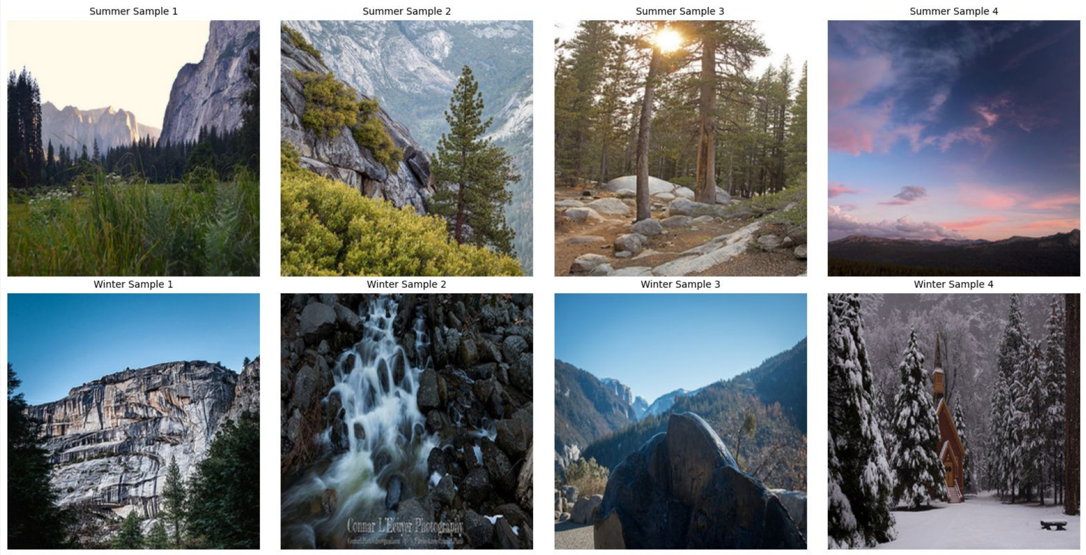
2. Summer: Images containing summer scenes characterized by:
   - Green vegetation
   - Warm weather and sunny skies
   - Beaches and water activities
   - Summer clothing
   - Summer activities

Data Collection Process

Dataset :

- **Initial Dataset:**
  - 2,740 total images
  - Training: 2,193 images (1,231 summer + 962 winter)
  - Test: 547 images (309 summer + 238 winter)
- **Training Set:** 600 images (300 per class)
- **Validation Set:** 100 images (50 per class)
- **Total:** 700 images
- **Source:** Kaggle (Summer2Winter Yosemite dataset) https://www.kaggle.com/datasets/balraj98/summer2winter-yosemite

Data Examples:



Data Collection Steps

1. Initial Acquisition

- Downloaded dataset from Kaggle, was organized into train/test splits

2. Data Reorganization

- Restructured for fastai compatibility
- Created organized directory structure

3. Data Cleaning

- Removed 148 converted or mismatched images to keep the dataset consistent.
- Used date metadata in filenames to determine the true season of each image.
- Compared this with the folder label (e.g., winter, summer).
- Images with season mismatches were identified as converted or artificially generated, so they were removed.

4. Random Sampling

- Applied random sampling for balance
- Final dataset: 300/300 training, 50/50 validation per class

Difficulties Encountered

Converted Images and Data Quality

- During exploration, it became clear that the dataset contained a number of artificially generated or converted images—for example, photos that had been stylistically altered or season-shifted using image conversion tools. These images introduced misleading visual cues and could negatively impact the model's ability to learn real seasonal patterns.
- **Solution:**
  To ensure data integrity, I extracted the date metadata encoded in each filename and used it to infer the true season the image should represent. Any image whose inferred season did not match its folder label was flagged as synthetic or incorrectly converted and removed from the dataset.

Initial Overfitting and Model Adjustments

- During the first training runs, the model showed clear signs of overfitting: the training loss continued to decrease while the validation loss began to rise. This behavior was likely caused by the relatively small dataset and the model's ability to memorize training examples rather than generalize.
- **Solution:**
  To address this, I enhanced the model architecture and training setup by introducing stronger data augmentation, increasing regularization, and applying dropout. These changes increased input variability and reduced over-reliance on specific features, resulting in more stable validation performance.

Model Architecture and Training Strategy

Architecture – ResNet34:
Used a 34-layer convolutional neural network with residual connections. This architecture was chosen because it offers a strong balance between model complexity and computational efficiency, making it well-suited for medium-sized image datasets.

Transfer Learning (Two-Stage Training):
Stage 1: The pre-trained backbone was frozen, and only the newly added classification head was trained. This allows the model to adapt quickly to the new task without disrupting the robust low-level features learned from ImageNet.
Stage 2: The entire network was unfrozen and fine-tuned end-to-end, enabling the model to specialize more deeply in distinguishing seasonal visual cues.

1cycle Learning Rate Policy:
Used a cyclic schedule with a base learning rate of **1e-4**, helping the model converge faster and avoid shallow minima.

Regularization:
Weight Decay (0.1): Applied L2 regularization to reduce overfitting and encourage smoother weights.
Dropout (0.5): Added dropout in the classification head to improve generalization by reducing reliance on specific activations.

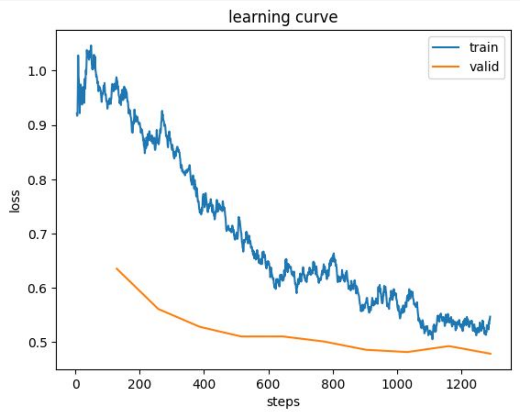
Early Stopping:
Training was halted if validation performance didn’t improve for **3 consecutive epochs**, preventing unnecessary training and overfitting.

Data Augmentation:
Applied strong augmentations—rotation, zoom, and lighting adjustments—to increase dataset variability and improve robustness to real-world visual conditions.

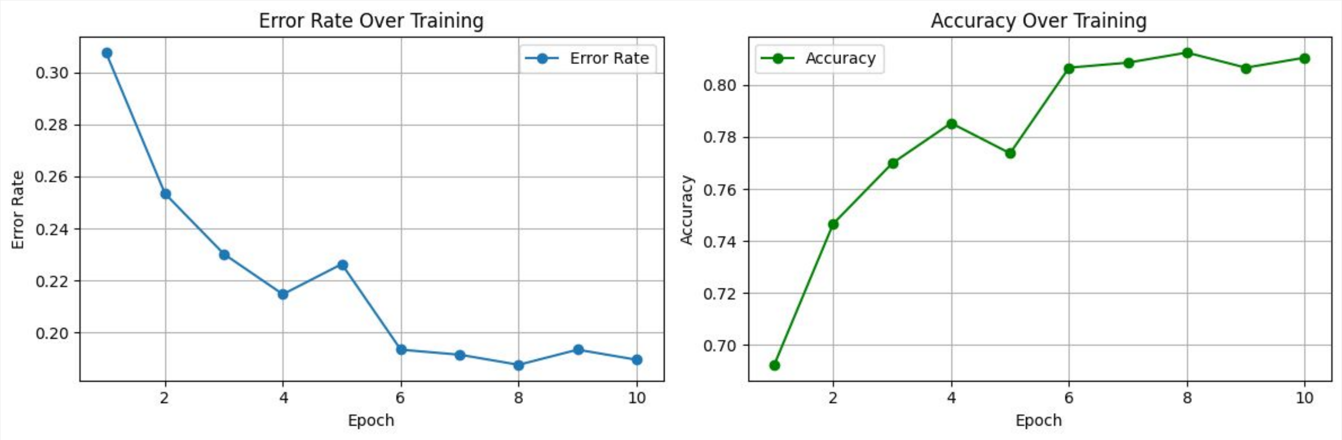
Results

Training progress, shows steadily decreasing training and validation loss as well as improving performance across epochs.
The error rate consistently drops while accuracy rises, indicating effective learning and good generalization.

Learning Curve



Error Rate and Accuracy

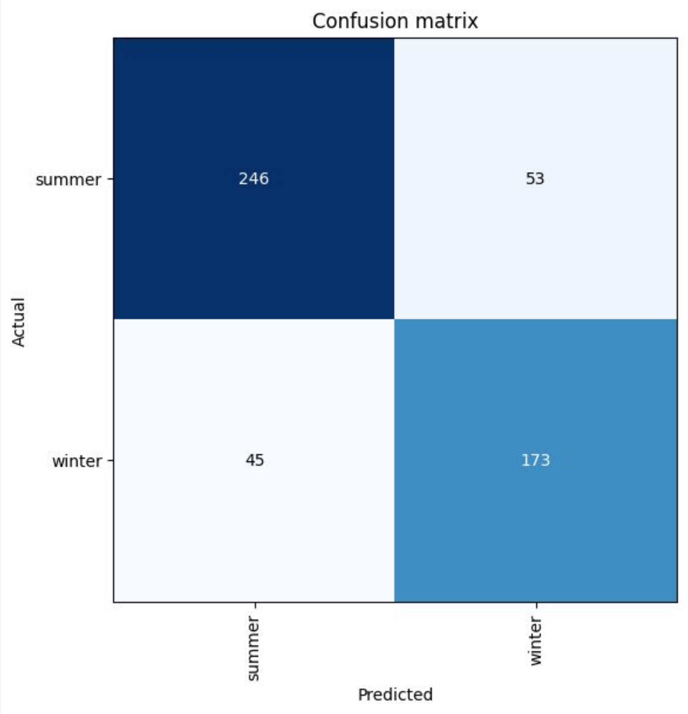


Results

Results: Achieved 81% accuracy on validation set (100 images)

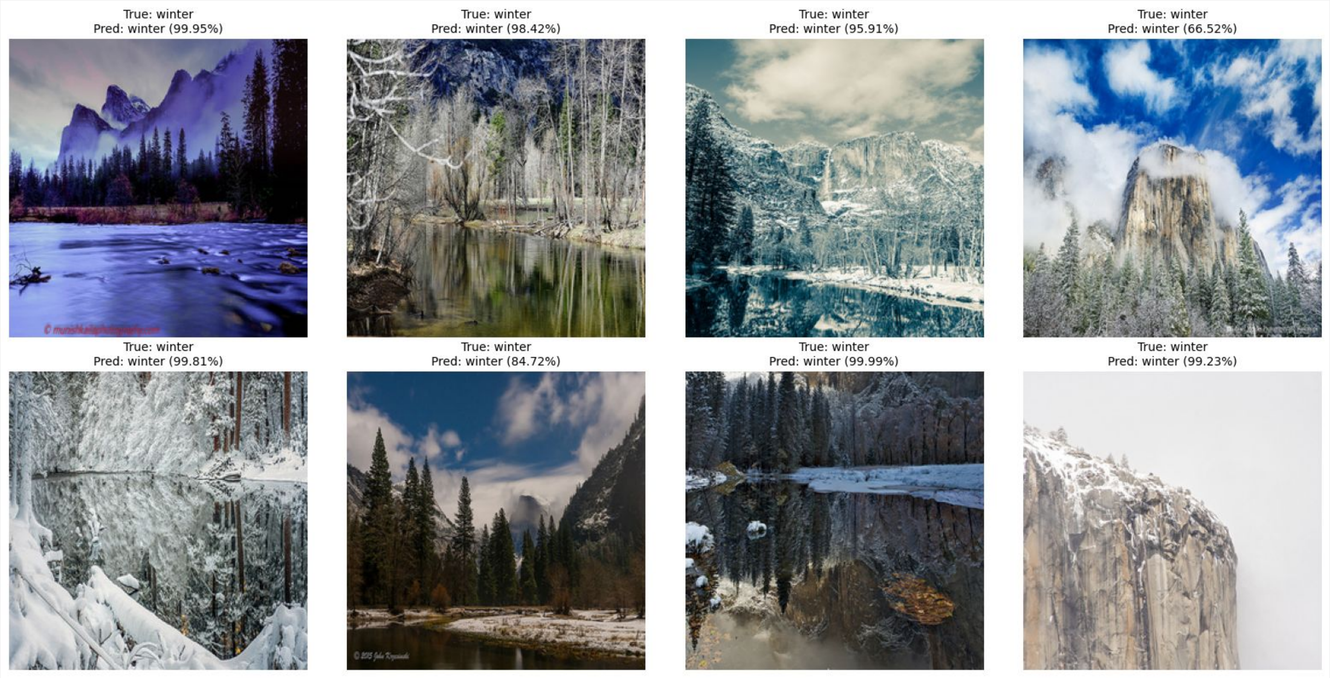
- **Summer class:** 84.5% precision, 82.3% recall
- **Winter class:** 76.6% precision, 79.4% recall
- In general, the model performs well, achieving roughly 80% accuracy even on a relatively small and somewhat ambiguous dataset.
- The model performs slightly better on summer images, likely due to more distinctive visual features.

```
Overall Accuracy: 0.8104 (81.04%)

Error Rate: 0.1896 (18.96%)

Detailed Classification Report:
              precision    recall  f1-score   support

      summer     0.8454    0.8227    0.8339       299
      winter     0.7655    0.7936    0.7793       218

    accuracy                         0.8104       517
   macro avg     0.8054    0.8082    0.8066       517
weighted avg     0.8117    0.8104    0.8109       517
```



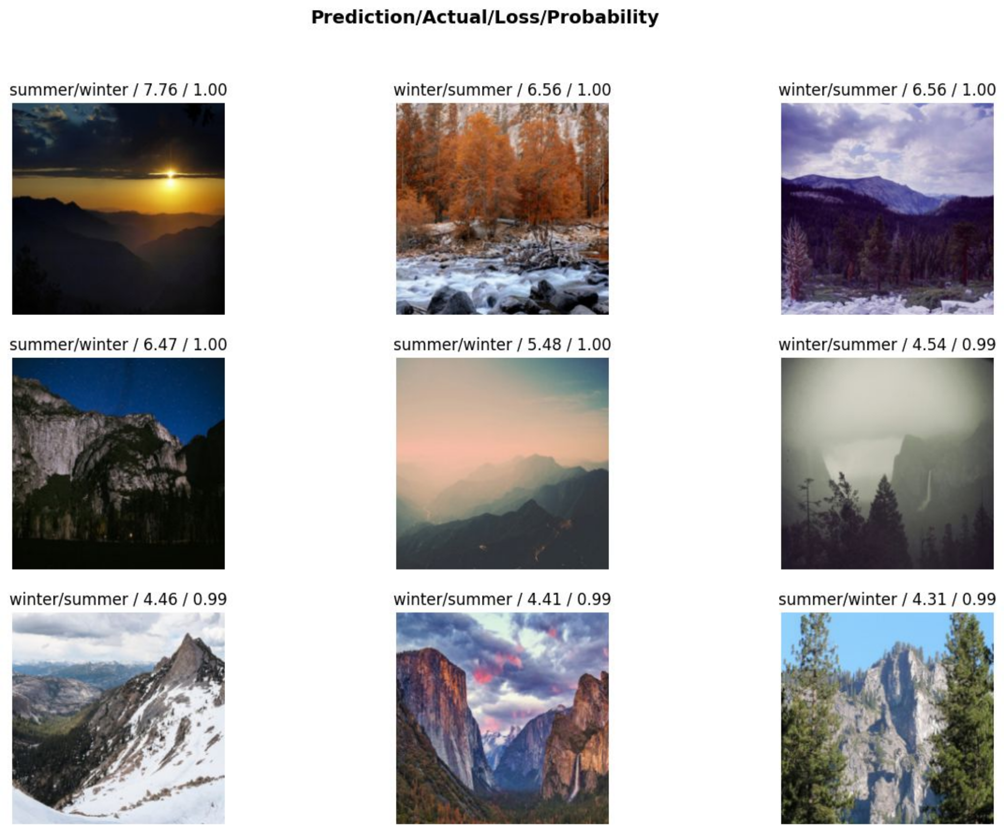
Examples of Correct Predictions

It can be seen that the model easily identifies clear winter scenes, especially those featuring snow and ice.



Top Losses Analysis

However, as expected the model struggles when images lack clear seasonal cues, leading to confusion in more non-specific cases like those shown here.



Potential Improvements

- **Data Augmentation:** Could add more diverse augmentation strategies
- **More Training Data:** Especially for edge cases and ambiguous scenes
- **Architecture:** Could try deeper models (ResNet50, ResNet101) or different architectures
- **Class Balancing:** Ensure equal representation of both classes
- **Data Cleaning:** Remove ambiguous or incorrectly labeled images

Link to the Project

https://github.com/NaamaAvn/deep-learning-ex1/blob/main/project.ipynb