Nahal Bagheri
CS-171
Project Proposal

Title: Global Natural Scene Recognition

Introduction:

- The realm of image classification has expanded significantly with the advent of convolutional neural networks (CNNs), offering unprecedented precision in interpreting and categorizing visual data. My project, "Global Natural Scenes Recognition," aims to harness the sophisticated pattern recognition capabilities of CNNs to identify and categorize images of natural landscapes from around the world. This task holds substantial importance in various applications, such as geographic information systems, environmental monitoring, and even in enhancing user experience in travel and educational applications.
- The machine learning task at hand is not merely an exercise in technological prowess but a step towards deeper machine understanding of our natural world. Through accurate classification of scenes such as seas, forests, mountains, and more, we can facilitate a better interaction between AI and the environment, aiding in efforts ranging from ecological conservation to the curation of geo-specific content for educational platforms.
- For this project, I have chosen a robust dataset from Kaggle, renowned for its diversity and quality of image data. The dataset contains thousands of labeled images across multiple categories of natural scenes, providing a rich training ground for our CNN model. The dataset is instrumental for the development and fine-tuning of our algorithm, ensuring that my model can generalize well across various types of natural landscapes.

Problem Statement:

The core problem this project aims to solve is the accurate classification of natural scene images into their respective categories using neural networks. Unlike controlled environments, natural scenes are subject to a multitude of variables such as lighting, weather, seasonal changes, and occlusions, which add layers of complexity to the classification task. The goal is to develop a neural network that can robustly discern and categorize images into classes like seas, forests, mountains, etc., despite these challenges.

Real-world Implications:

The implications of solving this problem are far-reaching. An accurate natural scene classification system can significantly enhance environmental monitoring by enabling the automated analysis of land use and land cover. It could aid climate change research by tracking changes in natural habitats and assessing the health of various ecosystems. In the consumer space, such a system could improve the curation of content for social media platforms and travel apps, where users often search for natural landscapes and sceneries. Moreover, the advancements in this area could contribute to the fields of autonomous navigation and Earth observation, providing valuable data for educational, commercial, and scientific applications. The success of this project could mark a step forward in our ability to understand and interact with the natural world through the lens of technology.

Data Description:

The dataset chosen for this project is the "Intel Image Classification" dataset, available on Kaggle at the following link: Intel Image Classification Dataset. This dataset has been compiled and curated for image classification benchmarking, specifically designed for natural scene recognition tasks.

Key Features and Labels:

- The dataset comprises approximately 25,000 images of various natural scenes.
- The images are color images with a resolution of 150x150 pixels, stored in JPEG format.
- They are divided into six categories, serving as labels for the classification task: buildings, forest, glacier, mountain, sea, and street.
- The dataset is prearranged into 'train' (approximately 14,000 images), 'test' (approximately 3,000 images), and 'pred' (unlabeled images intended for predictions, approximately 7,000 images) sets.

Preprocessing Steps Anticipated:

- Data Cleaning: While the dataset is prearranged and categorized, a preliminary inspection of the images will be necessary to ensure there are no mislabeled or irrelevant images that could impair the training process.
- Feature Engineering: Given that the images are already sized uniformly, additional feature engineering might not be essential. However, augmentation techniques like rotation, zooming, and flipping could be applied to increase the dataset's robustness and help prevent overfitting.
- Handling Missing Values: In image datasets, 'missing values' typically aren't a concern in the traditional sense. However, any corrupt images that do not load correctly will need to be identified and removed from the dataset.
- Normalization: It is standard practice to normalize image data. This process involves scaling pixel values to a range of 0 to 1 to help the convergence of the network during training.

- Splitting the Dataset: The dataset is already split into training, testing, and prediction sets. It would be important to maintain this division to properly train and validate the model's performance.
- Label Encoding: The categorical labels will be encoded into numerical form, which could be one-hot encoding or integer encoding, depending on the chosen model architecture and loss function.

By addressing these preprocessing steps, the dataset will be adequately prepared to train a neural network for the classification of natural scenes.

Methodology:

The methodology for this project will primarily revolve around Convolutional Neural Networks (CNNs), given their proven efficacy in image recognition tasks. CNNs are particularly adept at preserving the spatial hierarchy of features in an image, making them ideal for this application.

Neural Network Architecture and Configuration:

Input Layer: The input layer will accept color images with the shape of 150x150x3, corresponding to the image dimensions and the three color channels (RGB).
Convolutional Layers: Multiple convolutional layers will be used, with varying numbers of filters starting from 32 in the initial layers and increasing in deeper layers. These layers will be responsible for feature extraction.
Activation Functions: Each convolutional layer will be followed by a ReLU (Rectified Linear Unit) activation function to introduce non-linearity into the model, allowing it to learn more complex patterns.
Pooling Layers: Max pooling layers will be interspersed between convolutional layers to reduce dimensionality and computational load, while preserving the most salient features.
Dropout Layers: Dropout will be employed after pooling layers to prevent overfitting by randomly setting a fraction of input units to 0 at each update during training time.
Flattening Layer: After several convolutional and pooling layers, the high-level reasoning in the neural network is done via fully connected layers. The multi-dimensional output will be flattened into a one-dimensional array for processing in the fully connected layers.
Fully Connected Layers: One or more fully connected layers will follow the flattening step, leading toward the output layer. These will be crucial for combining features and making classification predictions.
Output Layer: The output layer will consist of neurons equal to the number of classes (six for this dataset), with a softmax activation function to output a probability distribution over the classes.

Specific Algorithms, Libraries, or Frameworks:

- The project will leverage TensorFlow and Keras, which are powerful and widely used libraries in the machine learning community for building and training neural network models. TensorFlow provides the backend computational engine, while Keras offers a user-friendly interface to construct the neural network layers.
- To improve the model's accuracy and generalization, techniques such as data augmentation and transfer learning using pre-trained models like VGG16 or ResNet may be explored.
- The training process will involve the use of optimization algorithms such as Adam or Stochastic Gradient Descent (SGD) with momentum, which are effective for navigating the high-dimensional weight space of a neural network.
- Early Stopping and Model Checkpointing will be utilized during training to monitor the model's performance on a validation set and to save the best model for later use.

By employing this CNN architecture and leveraging the mentioned frameworks and techniques, the project aims to build a robust model capable of classifying natural scene images with high accuracy.

Innovation:
This project introduces innovation through the integration of ensemble learning techniques with deep CNNs to enhance classification accuracy, a novel approach that combines the hierarchical feature extraction prowess of CNNs with the decision-making robustness of algorithms like Random Forests.

Objectives:
The primary objective is to achieve superior classification performance on the natural scenes dataset, setting a benchmark in precision and recall metrics. Additionally, the project aims to demonstrate the practical feasibility of deploying such a model in real-world applications, such as environmental monitoring and content curation platforms.

Evaluation Metrics:

Accuracy: This is the most straightforward metric, indicating the overall proportion of correct predictions. It provides a quick understanding of the model's effectiveness across all classes.
Recall: Also known as sensitivity, recall calculates the ratio of true positives to the sum of true positives and false negatives. This metric is important when the ability to find all positives (all instances of a particular class) is important.

F1-Score: The F1-score is the harmonic mean of precision and recall. It's a better measure than accuracy for imbalanced datasets and is useful when you need a balance between precision and recall.

Relevance:

These metrics are relevant for the natural scene classification problem because we need to ensure not only overall high performance (accuracy) but also that the model is equally good at identifying each specific class (recall), given the potential imbalance in the number of images per category. The F1-score will be particularly useful to gauge the model's performance in a more balanced manner, taking both false positives and false negatives into account, which is critical for applications where misclassification can have significant consequences.

Related Work:

The field of image classification has seen numerous studies and projects, especially using CNNs, due to their state-of-the-art performance on visual recognition tasks. Notable research has been published on the classification of natural scenes, where models like AlexNet, VGGNet, and ResNet have been trained and validated on datasets similar to the one chosen for this project. These studies have showcased high levels of accuracy in distinguishing between different natural environments.

However, a common limitation in existing approaches is the reliance on large, complex models that require substantial computational resources, making them less accessible for real-world applications with limited hardware. Additionally, while accuracy is often high, there remains room for improvement in the precision and recall for certain classes, which can be critical in applications like environmental monitoring where misclassification can have significant consequences.

The project aims to address these gaps by exploring more computationally efficient models that do not compromise on classification performance and by employing ensemble techniques to improve the precision and recall across all classes. This approach seeks to make the solution more viable for deployment in varied real-world scenarios.

Timeline:

Week 1: Data Preprocessing, Model Development and Training

Week 2: Ensemble Learning Integration and Further Training, Evaluation and Optimization

Week 3: Documentation and Reporting, Project Submission

Conclusion:

This project proposal outlines a comprehensive plan to leverage Convolutional Neural Networks (CNNs) and ensemble learning methods for the classification of natural scene images. I will utilize a well-structured dataset from Kaggle, which includes diverse images of natural landscapes categorized into six classes. The methodology described aims to overcome limitations of current approaches by proposing a more computationally efficient model without sacrificing accuracy.

The innovative aspect of this project lies in combining CNNs with ensemble learning techniques, aiming to enhance the precision and recall of the classification task, especially in real-world applications where resources may be limited. I intend to employ popular machine learning libraries such as TensorFlow and Keras, and I will evaluate my model using metrics that give a rounded view of its performance, such as accuracy, precision, recall, and the F1-score.

By addressing the gaps in existing research, particularly the need for resource-efficient models with high classification performance, this project has the potential to impact various fields, from environmental monitoring to enhancing user experience in digital applications. The anticipated outcome is a robust, reliable model that sets a benchmark in natural scene recognition and can be applied to solve practical problems in a global context.

References:

1. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems, 25.
2. Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv:1409.1556 [cs].
3. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
4. Intel Image Classification. Kaggle. Retrieved from https://www.kaggle.com/datasets/puneet6060/intel-image-classification.
5. Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. O'Reilly Media.

Introduction:

Dataset, Website:
CAD Cardiac MRI Dataset (kaggle.com)

List of References

[1] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5686931/#:~:text=other%20epidemiological%20studies.-,The%20major%20risk%20factors,activity%3B%20unhealthy%20diet%20and%20stress