ONTARIO TECH UNIVERSITY
DEPARTMENT OF COMPUTER SCIENCE

Group 1 December 8, 2024

LOL: Landmarks Organized by Labels

by

Adam Orfao Aria Sinaei Faraj Al-Hussaini

Abstract: Diversity and complexity of real world imagery is high, and recognizing specifics in images is no small feat. In this project, we developed LOL: Landmarks Organized by Labels. Aimed to both adapt and test several fine-tuned models and their relative aptitude in terms of landmark recognition. Utilizing a vast dataset provided by Google and hosted by Wikipedia, several different image processing deep learning models were trained and tuned. Iterating through new models, and differently portioned datasets, the overall accuracy was increased with each major iteration we made, totaling 3. The final model output a testing accuracy of 71.29% out of 78 major hierarchical categories describing the type of landmark, and a 93.44% accuracy discerning whether the presented landmark is natural or man-made. Saving the model, it was then hooked into a simple python program that allows uploaded images to then be predicted on. This project highlights the importance of iteration and improvements in numerous aspects in order to reach a greater final outcome, especially in complex deep learning.

1 Introduction

Recognizing landmarks from images is a crucial application of computer photography and computer vision. Having applications in tourism, navigation, and further cultural implications. Landmark recognition also poses a significant challenge as an individual landmark can be represented in numerous ways. Different angles, lighting, and perspectives fundamentally change the way an image is laid out and, as a result, can be very difficult to correctly and consistently classify. Using deep learning has shown promise in addressing issues like these, but with the sheer amount of data and computational time, there still remain challenges in addressing this ever-present dilemma.

This project, Landmarks Organized by Labels (LOL), aims to get a more comprehensive look at how different models and data shape the overall accuracy of landmark recognition. It also aims to see, on a smaller scale, how well models gain information using different methods. Using a public data set provided by Google, various samples were taken and then used to tune and train the models. These models were then used in order to attempt to classify images uploaded by the user, through a simple python interface.

Our focus is primarily on the iterative development process of the recognition model. Across three primary iterations, we experimented with several architectures for the model and strategies to extract the data effectively. The final output for this project was integrated and exported into a simple application providing a GUI to upload images, preview them, and view the model's predictions.

This paper presents both the methods and results of our project, including the insights gained from tuning and trading models, and the trade-offs made to use the data and computational power we had access to effectively. Through continuing to explore the learning of models across iterations, this paper aims to contribute to the broader understanding of effective practices in both machine learning, data mining and sampling, as well as image recognition and classification. [1]

[1] Project Repository: `https://github.com/oadam03/LOL-landmark-recognizer`

2 Methodology

This project utilized three versions of a multi-task learning pipeline to classify landmark images based on hierarchical labels and whether they are natural or human-made. Each version introduced refinements and enhancements over the preceding one to improve data pre-processing, balance, and model training. Below is a detailed breakdown of the methodologies for each version:

2.1 Version 1

1. **Data Preparation:**
    - Merged two datasets (`train.csv` and `train_label_to_hierarchical.csv`) to include essential columns like `landmark_id`, `hierarchical_label`, and `natural_or_human_made`.
    - Filtered rows with missing values for `hierarchical_label` and `natural_or_human_made`.
    - Retained only images present in the local directory and sampled a subset of 3,050 rows (images) for training.
2. **Data Encoding:**
    - Encoded categorical columns `hierarchical_label` and `natural_or_human_made` using label encoding to prepare them for training.
3. **Model Architecture:**
    - Used a ResNet-50 pre-trained model, replacing its fully connected layer with two task-specific heads for hierarchical and natural/human-made classifications.
    - Defined a multi-task learning model that shared a base architecture but had separate output layers for the two tasks.
4. **Training and Evaluation:**
    - Implemented a training loop with cross-entropy loss for both tasks and utilized an Adam optimizer for parameter updates.
    - Split the dataset into training, validation, and testing sets with a 70-20-10 ratio.

2.2 Version 2

1. **Dataset Improvements:**
    - Addressed class imbalance by applying a custom subsampling strategy. This strategy combined equal and proportional distribution weights to ensure better representation for underrepresented classes.
    - Saved a balanced dataset as `balanced_train_v2.csv` for consistent usage.
2. **Visualization and Analysis:**
    - Analyzed the distribution of hierarchical labels to identify skewness in the dataset.
    - Visualized both the original and balanced label distributions using bar plots to validate the effectiveness of the balancing strategy (Figures 4 and 5).
3. **Model Adjustments:**
    - Refined the training pipeline to handle balanced data.
    - Enhanced the DataLoader to filter invalid data and improve efficiency in processing.
4. **Enhanced Data Handling:**
    - Incorporated additional checks to ensure all images in the dataset were accessible and valid before encoding and transformation.
    - Introduced parallelized data downloading for scalability using asynchronous processing (commented out in the final version for simplicity).

2.3 Version 3

1. **Multi-Task Learning Enhancements:**
    - Introduced a more advanced data preprocessing pipeline, streamlining image normalization and transformation.
    - Optimized hyperparameters for training, including learning rate and batch size.
2. **Architectural Refinements:**

- Improved the base ResNet-50 model's feature extraction pipeline by fine-tuning layers specifically for hierarchical and natural/human-made tasks.
- Utilized distinct evaluation metrics for each task to better capture model performance (details of these metrics are deferred to the results section).

3. **Training Stability:**
   - Implemented robust error handling during training and evaluation to manage missing or corrupt images gracefully.
   - Optimized the data loading process to minimize bottlenecks and ensure smooth batch processing.
4. **Comprehensive Documentation:**
   - Introduced structured logging and summaries at each step to facilitate debugging and traceability.

Each version of the methodology built upon the previous, addressing limitations and introducing novel enhancements to ensure a robust and scalable pipeline for the classification tasks. This iterative approach underscores the importance of refining data handling, model architecture, and training protocols to achieve more reliable results.

3 Results

The results are outlined across the three versions of the pipeline, highlighting the improvements and changes in performance metrics for hierarchical classification (H-Acc) and natural vs. human-made classification (N V.S H Acc).

3.1 Version 1

- **Dataset Size:**
  - Train: 1,757 images
  - Validation: 502 images
  - Test: 251 images

- **Training Performance:**
  - Hierarchical Accuracy: 40.81%
  - Natural vs. Human-Made Accuracy: 91.52%
- **Test Performance:**
  - Hierarchical Label Accuracy: 24.70%
  - Natural vs. Human-Made Accuracy: 84.46%
- **Key Notes:**
  - Initial performance was limited by the small dataset size and random sampling.
  - Observed the need for cleaning and balancing data.
  - Recommended proportional sampling for better representation across 78 hierarchical categories.

3.2 Version 2

- **Dataset Improvements:**
  - Balanced sampling strategy applied to address skewed data representation.
  - Dataset size remained at approximately 2,500 images, distributed proportionally.
- **Training Performance:**
  - Hierarchical Accuracy: 38.40%
  - Natural vs. Human-Made Accuracy: 87.63%
- **Test Performance:**
  - Hierarchical Label Accuracy: 38.40%
  - Natural vs. Human-Made Accuracy: 87.63%
- **Key Notes:**
  - Improved hierarchical accuracy due to balanced data.
  - The model's ability to differentiate between natural and human-made landmarks showed slight improvement.

  - Still constrained by dataset size and architecture limitations.

3.3 Version 3

- **Dataset Size:**
  - Train: 10,997 images
  - Validation: 3,142 images
  - Test: 1,571 images
- **Model Enhancements:**
  - Switched to EfficientNet-B3 from ResNet50 for better feature extraction.
  - Image size increased to 300x300 pixels for enhanced detail capture.
  - Training epochs doubled from 10 to 20.
  - Batch size reduced to 16 to accommodate computational limitations.
- **Training Performance:**
  - Hierarchical Accuracy: 91.18% (Epoch 20)
  - Natural vs. Human-Made Accuracy: 98.04% (Epoch 20)
- **Test Performance:**
  - Hierarchical Label Accuracy: 71.29%
  - Natural vs. Human-Made Accuracy: 93.44%
- **Key Notes:**
  - Significant improvements in both hierarchical and natural vs. human-made classification tasks.
  - The larger dataset and EfficientNet architecture contributed to the model's robustness.
  - High testing accuracy demonstrated reliable generalization across unseen data.

3.4 Summary

Across the three versions, the pipeline evolved significantly in terms of data quality, balancing, and model architecture. Version 3 outperformed the earlier iterations by leveraging a larger and better-balanced dataset, more advanced model architecture (EfficientNet-B3), and extended training epochs.

Version 3 is the clear winner, achieving:

- Hierarchical Label Accuracy: 71.29%
- Natural vs. Human-Made Accuracy: 93.44%

Table 1: Comparison of Model Versions

Version	**V1**	**V2**	**V3**
Model	ResNet50	ResNet50	EfficientNet-B3
Epochs	10	10	20
Dataset Size	2,500 images	2,500 images	16,000 images
Data Selection	Random selection	Weighted selection	Weighted selection
H Acc	24.70%	38.40%	71.29%
N V.S H Acc	84.46%	87.63%	93.44%

4 Conclusions

This project incrementally improved landmark recognition by refining data selection, balancing class distributions, and upgrading model architectures. Starting with small, imbalanced datasets and a ResNet-50 model, we progressed to a larger, more balanced dataset and an EfficientNet-B3 backbone. This approach increased hierarchical classification accuracy from 24.70% to 71.29% and natural vs. human-made classification accuracy from 84.46% to 93.44%. The results highlight the impact of careful data preparation, model tuning, and iterative development. Future work could involve expanding the dataset, exploring more advanced architectures, or integrating additional tasks, such as object detection or scene segmentation, to further enhance the model's ability to classify landmarks with greater precision and context.

5 Appendix: Program Demonstration

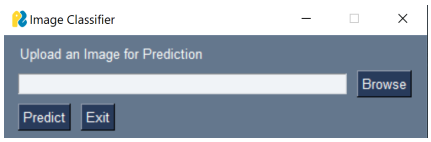

Figure 1: The GUI input screen for selecting an image to classify.

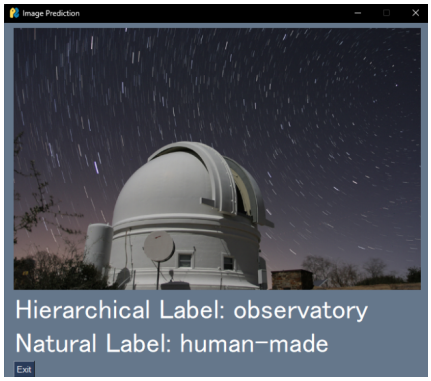

Figure 2: Output Example 1: The model displays predicted hierarchical category and natural/human-made classification.

Figure 3: Output Example 2: A different uploaded image and the corresponding classification results.

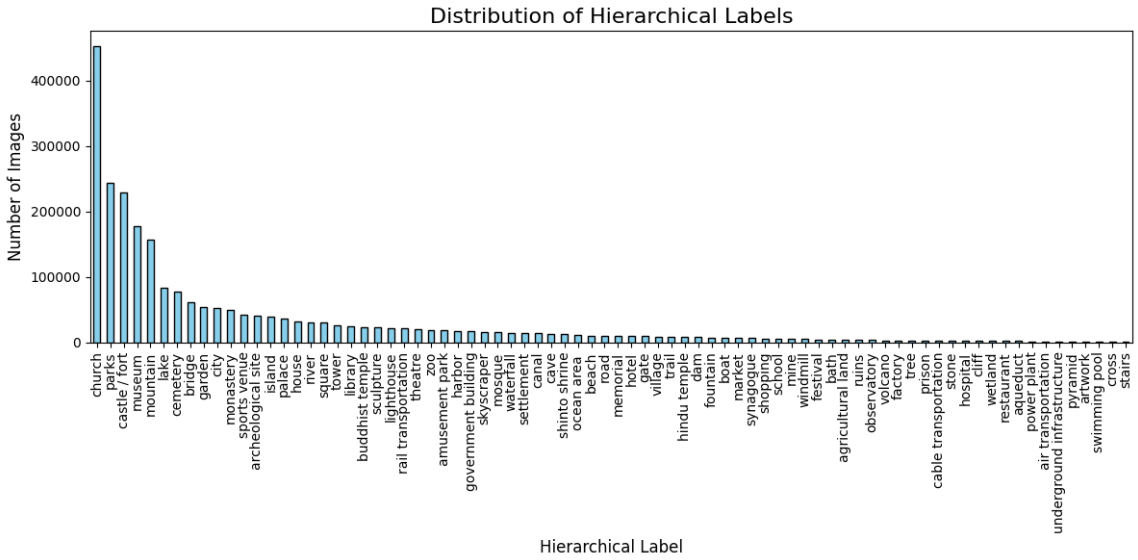

Figure 4: Graph 1: Number of Images V.S Hierarchical Labels Non-Balanced

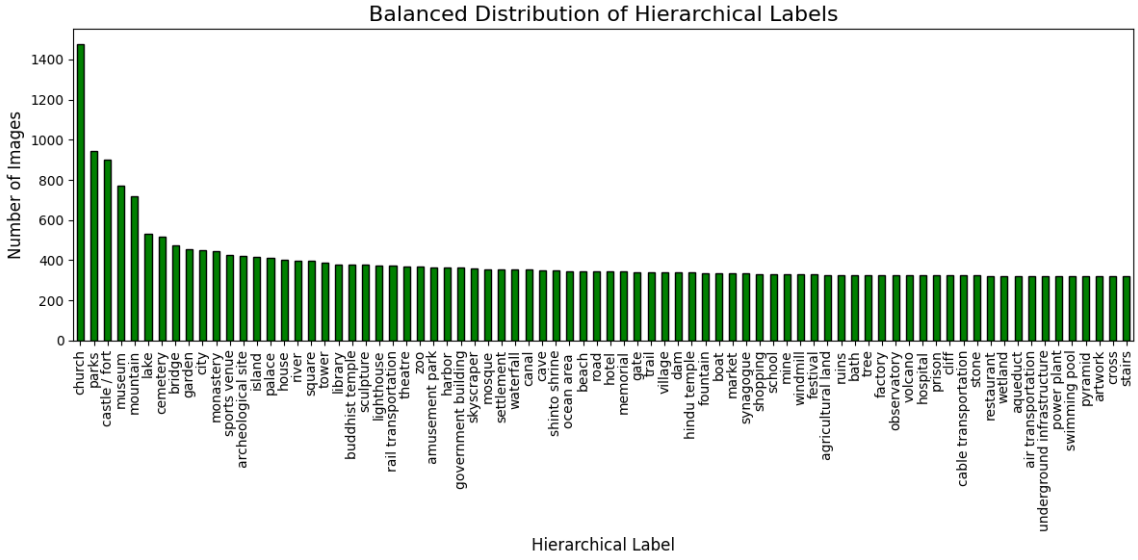

Figure 5: Graph 2: Number of Images V.S Hierarchical Labels Balanced

References

[1] E. Ramzi, N. Audebert, C. Rambour, A. Araujo, X. Bitot, and N. Thome, “Optimization of Rank Losses for Image Retrieval,” In submission to: *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep Residual Learning for Image Recognition. *arXiv preprint arXiv:1512.03385*. DOI: `https://doi.org/10.48550/arXiv.1512.03385`.

[3] T. Weyand, A. Araujo, B. Cao, and J. Sim, “Google Landmarks Dataset v2 - A Large-Scale Benchmark for Instance-Level Recognition and Retrieval,” *Proc. CVPR’20*, 2020.