2020 Fall COMP4471 Project Milestone
Google Landmark Recognition

LAM Man Hei
HKUST
mhlamaf@ust.hk

LIAO Yi Han
HKUST
yliaoag@ust.hk

1. Introduction

Deep convolution neural network has already gotten remarkable results in image classification tasks. In this project, we are going to investigate landmark recognition with deep convolution neural network, in which we will develop a landmark recognition model for people to better understand and organize their photo collections. We will attempt to use ResNet[2] and EfficientNet[4] to construct a landmark recognition model. Both model performance will be evaluated, and the best model will be elected as the final landmark recognition model.

1.1. Problem statement

The dataset used in this project is retrieved from Google Landmark Recognition 2020[1], which is a Kaggle competition. Since the competition is closed and no longer accept submission, the ground truth of the test dataset cannot be retrieved. Thus, only train dataset will be used in this project, and it is split into train, validation and test set. The expected result of the model is a column vector of the probability of each label. The label with the highest probability would be the predicted label of that image. To evaluate the performance of a model, some basic metrics are used, including accuracy, precision, recall, F1-score and confusion matrix.

2. Technical Approach

To achieve the objective, the following pipeline is proposed. First, exploratory data analysis (EDA) is done with the dataset to have a better understanding of the dataset. After that, some preprocessing work will be done based on the result of EDA. Then, the proposed model, ResNet and EfficientNet, will be trained and evaluated in order to find the best model for landmark recognition.

2.1. Exploratory Data Analysis

In exploratory data analysis (EDA), we would inspect the dataset and try to summarize the characteristics of the dataset, such as the property of the images and the balancedness of the dataset. Some graph visualization would be made, such as histogram and box plot, in order to get a better understanding of the dataset.

2.2. Preprocessing

In pre-processing, the images is first resize to 224 x 224 in order to fit into the models (ResNet and EfficientNet). Then, apply normalization to the images data. To increase the amount of data samples for those landmark classes with extremely limited samples, using data augmentation to enlarge their sample size. It also helps reduce the overfitting problem.

2.3. Landmark Recognition Model

In this project, two deep convolution neural networks, ResNet and EfficientNet are tested and evaluated. The models are implemented with PyTorch[3] along with some common python packages, such as numpy, pandas, and sci-kit learn. Since the dataset is very large, it is time-consuming to train a brand new network due to the limited hardware situation. Therefore, transfer learning technique is applied in order to shorten the training time. The final fully connected layer of the models is replaced with a new fully connected layer with the output size the same as the number of landmarks. Then, the model will be trained for a few epochs.

In the training progress, two loss functions, cross entropy loss and label-distribution-aware margin (LDAM) Loss[1] are tested. Cross entropy loss is a very common loss function used in classification problem. It can be considered as a baseline loss function. However, the dataset may have data imbalance issue, which cannot be solved by cross entropy loss. Using this loss function with a extreme imbalance dataset may cause low precision/recall in minority class. To get a better representation on minor class, LDAM loss can

[1] www.kaggle.com/c/landmark-recognition-2020

be use. The loss function is formulated as such:

$$\mathcal{L}_{LDAM}((x,y);f) = -\log \frac{e^{z_y-\Delta_y}}{e^{z_y-\Delta_y}+\sum_{j\neq y} e^{z_j}} \quad (1)$$

$$, \Delta_j = \frac{C}{n_j^{1/4}} \textbf{ for } j \in [1,k] \quad (2)$$

where C is a hyperparameter, n_j is the number of sample of j-th class, and z_j is the model prediction output of j-th class. By using LDAM loss, it attempts to increase the margin distance of the minority classes from the decision boundary.

In addition, hyperparameters tuning is required in order to obtain an optimized model. Grid search or Bayesian optimization, which is a more efficient technique, is applied to tune model hyperparameters.

2.4. Evaluation

To evaluate the performance of the deep learning model, accuracy, precision, recall and F1 score would be used.

$$\textbf{Precision} = \frac{TP}{TP+FP} \quad (3)$$

$$\textbf{Recall} = \frac{TP}{TP+FN} \quad (4)$$

$$\textbf{F1} = 2 \times \frac{\textbf{Precision} \times \textbf{Recall}}{\textbf{Precision} + \textbf{Recall}} \quad (5)$$

where TP, TN, FP, FN is the number of true positive, true negative, false positive, false negative prediction of the class respectively. Moreover, confusion matrix would be a visual evaluation metric.

3. Preliminary Results

3.1. Exploratory Data Analysis

There are 81313 landmark classes with 1580470 images in total.

Through EDA, an important feature of the dataset is detected that the dataset is very imbalanced as shown in figure 2. The largest landmark class sample size is 6272 while the smallest landmark class sample size is only 2. Almost 97.5% of the landmark classes are with sample size under 100.

Besides, the quality of images in the dataset is not very high. There are some quite misleading images even for human beings to recognize. For example, figure 1 shows how diverse the image in a landmark class can be.

3.2. Experiment

Before implementing the model on the whole dataset, top 12 and top 20 landmark classes is first selected to do some experiments. Each sub dataset is split into three parts: training, validation, and testing with the ratio of 0.72, 0.18, 0.1.

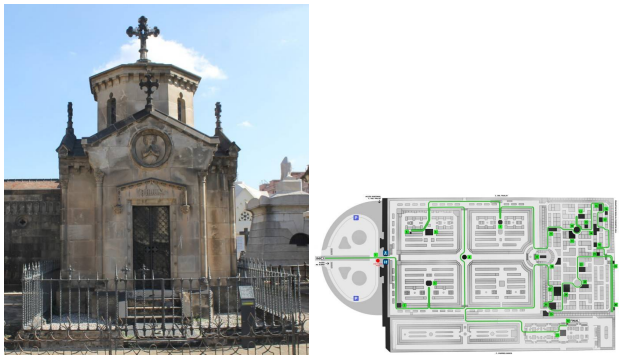

Figure 1. Example images of the same class in the dataset. The left image can clearly see the landmark where the right image is just a floor plan.

Figure 2. Frequency graph of each landmark class. The classes are sorted in descending frequency order.

Model	Training Accuracy	Testing Accuracy
ResNet-50	98.02%	96.65%
EfficientNet-B0	97.69%	98.05%

Table 1. Model performance of top 12 landmark classes dataset

For both datasets, using ResNet-50 and EfficientNet-B0 as training models and cross-entropy as loss function.

From the performance of these two sub-datasets(table 1 and table 2), we can observe that as more landmark classes are in the dataset, the accuracy will decrease. The main reason is that the minority class has relatively low precision and recall. As shown in the confusion matrix in figure 3, the image of landmarks with fewer samples are misclassified as some other landmarks with much more sample. This is a common issue in model training with an imbalance dataset. Therefore, to cope with the dataset with more landmarks and severe imbalance issues, LDAM will be applied to attempt resolving the issue in future works.

Model	Training Accuracy	Testing Accuracy
ResNet-50	96.05%	94%
EfficientNet-B0	92.40%	92.72%

Table 2. Model performance of top 20 landmark classes dataset

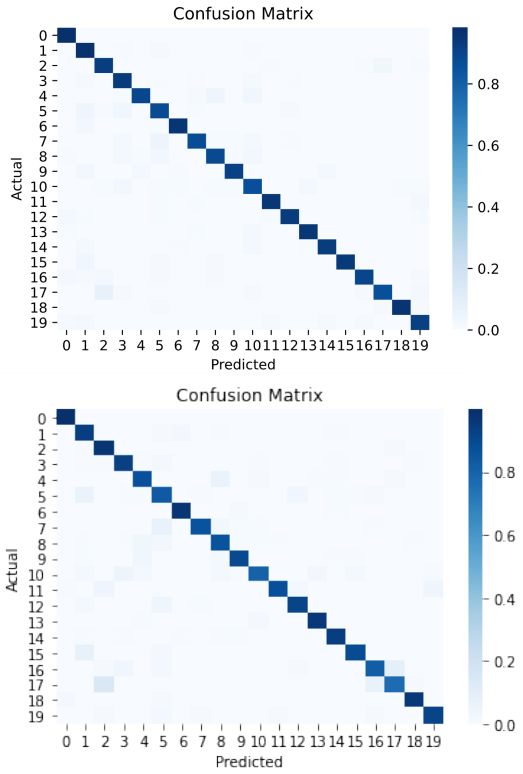

Figure 3. Test set confusion matrix of top-20 landmark classification with ResNet50(top) and EfficientNet-B0(bottom). The larger the landmark id, the fewer the sample that landmark have

References

[1] K. Cao, C. Wei, A. Gaidon, N. Arechiga, and T. Ma. Learning Imbalanced Datasets with Label-Distribution-Aware Margin Loss. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d. Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 1567–1578. Curran Associates, Inc., 2019.

[2] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. *arXiv:1512.03385 [cs]*, Dec. 2015. arXiv: 1512.03385.

[3] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *arXiv:1912.01703 [cs, stat]*, Dec. 2019. arXiv: 1912.01703.

[4] M. Tan and Q. V. Le. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *arXiv:1905.11946 [cs, stat]*, Sept. 2020. arXiv: 1905.11946.