Google Landmark Recognition

Siyuan Lin
Australian National University
u5794678@anu.edu.au

Shan Wang
Australian National University
U7094434@anu.edu.au

Zhengduo Zhu
Australian National University
U6302778@anu.edu.au

Abstract

When we browse the Internet, we are often attracted by some charming views. Sometimes people can recognize them quickly, but most of the time, people can not. Our challenge is to create a novel image recognition network that can help people recognize the landmark efficiently and correctly. In contrast to basic-level image recognition problem with limited class, the Google Landmarks Datasetv2 (GLDv2) [10] is a benchmark for large scale, fine-grained instance recognition in the domain of human made and natural landmarks. The GLDv2 contains more than 5 *million images and over* $200k$ *different classes. Additionally, the datasets are extremely long-tailed class distribution, a large fraction of non-landmark test photos and large intra-class variability. We introduced our model with the combination of non-landmark classifier module and multiple landmark classifiers module to solve this problem under limited classes and images due to the limited computational resources. In the non-landmark classifier, we use the ResNet [5] as backbone and deep neural network (DNN) to reduce feature dimension to speed up process time. In the multi-landmark classifier stage, we combine the efficientNet [9] and the Arcface loss [2] to solve large intra-class variability. The performance of our method can achieve* 0.25 *global average precision (GAP) within only* 10 *epochs.*

1. Introduction

Instance-level image recognition is a fundamental research topic that has been studied for decades. Due to the rapid improvement of the neural networks in recent years, the deep learning model achieved huge success theoretically and practically, including image recognition problems. For example, the ResNet [5], VGG [7] e.t.c shows strong performance in the basic-level image recognition on the ImageNet competition. The instance-level recognition, however, remains a challenge, especially under the large-scale domain. In the instance-level recognition, the model needs to recognize a specific instance of an object, rather than simply the category to which it belongs. Here, we use the GLDv2 [10] datasets as the benchmark for the google landmark competition on Kaggle. We propose a novel network that shows strong performance under only ten epochs training. Our network is mainly composed of two modules. To mimic the real-world scenario, there exist many non-landmark images in the GLDv2 [10] test datasets only. We use the non-landmark classifier with a pre-trained model from place365[11] to correctly label the out-of-domain images. In the second module, we use multi-landmark classifiers to extract each input image's global feature and analyze the output score. Due to the limited computational resources (Kaggle notebook has 9 hours limited in single training), we check our network's GAP within 10 epochs.

2. Related Work

2.1. Google Landmark Datasetv2

The Google Landmark Datasetv2 [10] aims to mimic real-world landmark recognition and retrieval problems. The GLDv2 [10] contains more than 5 million images and 200 thousand different classes. The datasets contain the following four major challenges. 1) Large scale: To cover the entire world-famous landmark, a corpus of millions of photos is necessary. Huge datasets often cause computational intensive. 2) Intra-class variability: The same landmark photos are taken under different angles or light conditions, even indoor and outdoor views. Even the same landmark may look differently from different angel or views. 3) Long-tailed class distribution: Some famous landmarks have more pictures (over 6000) than the less famous ones (less than 5) in the training stage. The model can be insensitive to those small training sample classes. The long-detailed distribution of the datasets can be viewed in Figure

1. 4) Out-of-domain queries: To simulate the real-world scenario, the search query may not be the landmark. There exist some non-landmark photos in the test datasets only. (Figure 4)

2.2. Google Landmark Recognition Competition

Existing methods for landmark recognition can be divided into three main groups, end-to-end method[1], Top-k re-ranking method[4] and the combination of global/local search the re-ranking method[3]. In global/local search and re-ranking method, use CNN model to find global features and then use another faster R-CNN model [3] to find local features in Top-k closest neighbors and a re-ranking step aims at distinguishing real landmark images from non-landmark images[4].Top-k re-ranking method is similar with global/local search and re-ranking method without using the local features search step[6]. End-to-end method further removed re-rank step and get the landmark ID and confidence by the classifier. In terms of trends, the method is getting simpler. With the help of increasingly powerful pre-trained networks, it is possible to solve the complex problems through a simple architecture. We adopted the end-to-end method. Instead of re-ranking step after landmark classifier we propose a non-landmark classifier in front. This method is also used by other groups but they separate landmarks to sub-classes such as building, tower, castle, sculpture, skyscraper, house, tree, watercraft, aircraft, fountain, etc. If the input image is not belong to them, it is a non-landmark image[1]. Because sub-classes is useless for our following landmark classifier, instead of sub-category classifiers, we added a binary classifier(non-landmark classifier) in front. It is inevitable to choose classes according to the number of images in the classes due to the massive number of classes and long-tailed distribution problems. We found other groups choose to use some classes to train the model, and extended the feature extraction function to all other classes. We believe for different classifier model will find different feature extraction function. Therefore we decided to train multiple landmark classifiers. About the backbone of our landmark classifier: The Efficient-Net[8], together with their open-source ImageNet pre-trained weights has been proven to be a state-of-the-art image classification model over the past year. We test Efficient-Net and ResNet as our backbone and found Efficient-Net perform better, so we decide to use Efficient-Net as our backbone. To solve large intra-class variability problem, [2] proposed an Additive Angular Margin Loss to get highly discriminative features for face recognition. We extended the method to landmark recognition.

3. Our Network

We now describe our network architecture in details, which is composed by the Non-landmark Classifier and Landmark Classifier 2 main modules.

3.1. Network Architecture

The detail of the structure of our method are shown in Figure 2. In the GLDv2 [10] test sets, there are approximately 98% test images are out-of-domain. Hence, we designed the front Non-landmark Classifier to distinguish the landmark and non-landmark input images. If the output of these classifiers is less than 50% we consider the test data is not a landmark image. If we can correctly identify non-landmark images, the test accuracy can be greatly improved. Since there are over $200k$ classes in the GLDv2 dataset [10] , it is impossible to fit them in one fully connected layer. So instead of using one landmark classifier, we trained multiple landmark classifiers. Our architecture is shown in figure2. We separate the landmark classifiers by numbers of images in each class. After analysis, we find that more than 12671 classes have more than 30 images, 14553 classes have more than 15 images and less than 29 images. Therefore, we choose 30 and 15 as a threshold. We did not train a model for classes less than 14 images due to the limited computational sources which means our model GAP can increase with more non-famous landmark classifier . To correctly assign the test image between classifiers, we normalize the output probabilities and output the maximum normalized probability and landmark ID. Normalize method is shown in equation1. Where p is the output probability of one landmark classifier; $ClassNum$ is a list of class numbers, length of the list the count of landmark classifiers.

$$p * ClassNum[current]/Max(ClassNum) \quad (1)$$

3.2. Non-landmark Classifier Architecture

In the non-landmark classifier shown in figure 2, we use the ResNet as our backbone. Following the global average-pool to extract the input image's global feature and the fully connected layer+softmax function to get final binary scores. Since the GLDv2 [10] contains no non-landmark images in the training stage, the transfer learning helps us refer to the same architecture pre-trained model knowledge from Place365 datasets [11] to classify landmark and non-landmark images.

3.3. Landmark Classifier Architecture

Now we detail the structure of the landmark classifier. In the landmark classifier shown in figure 3. We use Efficient-Net [9] as our backbone. Following the global average pool to obtain images' global feature, we add a DNN to further reduce the feature dimension to 1000. We then chose ArcFace loss [2] to alleviate the extreme intra-class variability problem. We replaced the fully connected layer with the

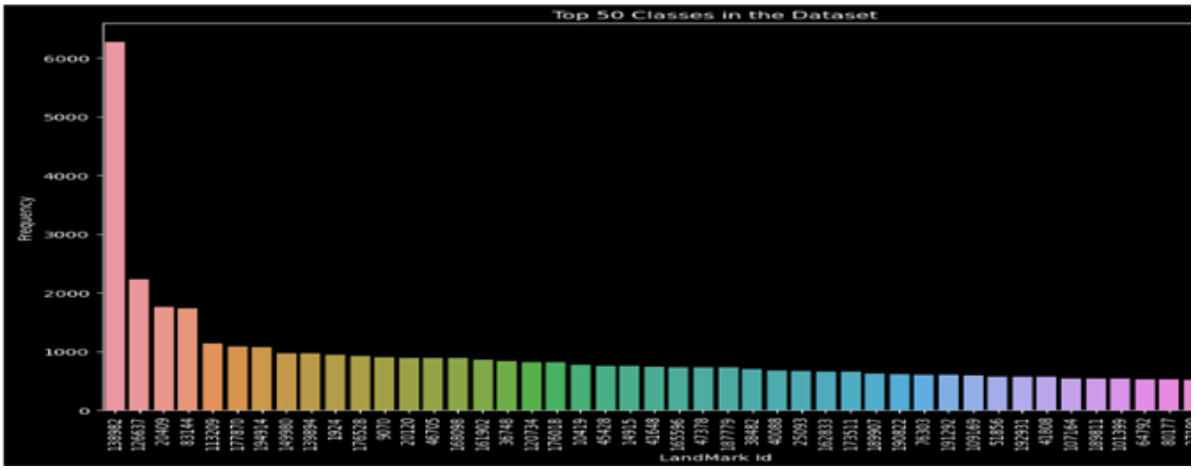

Figure 1. Long-tailed distribution of the GLDv2 [10] . In the figure, Famous landmarks own a dominant number of images, while the less-famous landmark has a limited number of images.

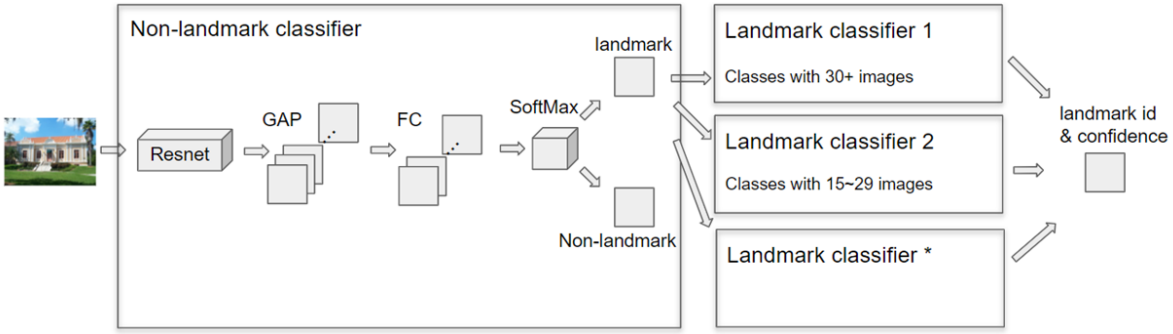

Figure 2. The architecture of our landmark recognition is composd by the non-landmark classifier and the multi-landmark classifier.

arc margin head. Arcface [2] optimizes the feature embedding to enforce higher similarity for intra-class samples and diversity for inter-class samples.

4. Experimental Result

4.1. Global Average Precision

The performance of our network will be evaluated by global averge precision (GAP). The GAP is computed as:

$$GAP = \frac{1}{M} \sum_{i=1}^{N} P(i) rel(i)$$

where:

- N is the total number of predictions returned by the system, across all queries
- M is the total number of queries with at least one landmark from the training set visible in it
- $P(i)$ is the precision at rank i
- $rel(i)$ denotes the relevance of prediciton i: it's 1 if the i-th prediction is correct, and 0 otherwise

4.2. Test platform

Due to time constraints, we choose to train locally and online. Our project does not consider the length of training time, but focuses more on the presentation of training effects. So online training and local training use different graphics cards.

The online training of this project uses the Kaggle platform. Kaggle provides us with access to NVIDIA TESLA P100 GPUs with GPU memory 16GB. The local graphics card is GTX TITAN X, the dedicated GPU memory is 12GB and the shared GPU memory is 16GB.

In order to compare the convergence of different network architectures vertically, we trained four network architectures: ResNet, EfficientNet, ArcFace+EfficientNet,

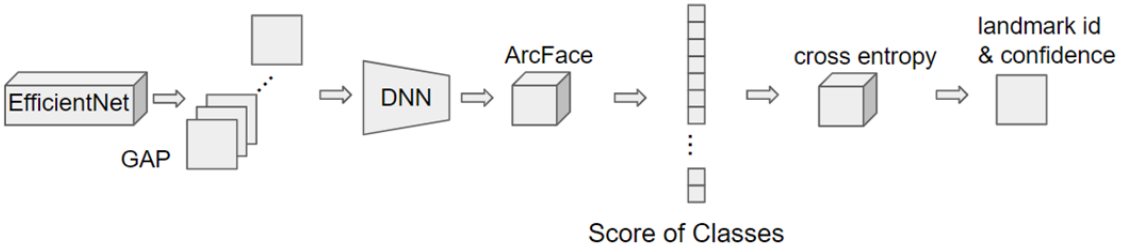

Figure 3. The architecture of the multi-landmark classifier.

Figure 4. Out-of-Domain Images

Non-landmark+ArcFace+EfficientNet. After training for 10 epochs and comparing their GAP, we can know which model can handle such a huge amount of data and classes.

4.3. Training result

4.3.1 Result of Non-landmark Classifier

Images in figure4 are classified as non-landmark images by this classifier. The results seems reasonable.From the image point of view, most of the photos of independent people are classified as non-landmarks. For the two-classes classification, there are a large number of datasets to provide enough training times. So the result is more accurate.

4.3.2 Cross Verify of Multiple Landmark Classifiers

We use cross verify to verify landmark have been correct assigned between different classifier models. We choose classes base on the number of images in that classes during data loading. We load verify images of one classifier and feed them into both classifier1 and classifier2. Compare the output probabilities after normalise it with number of classes in that classifier. We found more than 60% images have been correct assigned. But as mentioned above, the number of images in each class is extremely unbalanced. Therefore, in the course of the experiment, although classifier 1 and classifier 2 can process image classification more accurately, the global accuracy is still not very high. We will give specific values below.

4.3.3 Result of Landmark Classifier

First, we need to select the backbone of the model. Due to time factors, we only tested two basic models, ResNet18 and EfficientNet [9]. As a classic model, ResNet can handle image classification problems well, and at the same time, it can also detect some of the more difficult features. EfficientNet [9] expands the model in three dimensions through the compound model expansion method and neural structure search technology. After training, theoretically, it can be applied to various special situations. So we built the model on these two backbones, trained for 10 epochs, and recorded their GAP. The results are shown in the table1 below.

It can be seen from the table that in the first three epochs of training, EfficientNet [9] has a very big advantage, and the average difference can reach 0.022. Then, ReseNet [5] has a very stable rise, and finally reaches 0.1336. Efficientnet [9] has fluctuation in the fourth epoch. From 3 to 4 epoch of EfficientNet [9], it can be seen that the value of GAP has not changed significantly, and even dropped slightly. From 4 to 5, the GAP value has risen significantly and is flat with ResNet. The curve behind it is similar to ResNet[5]. Eventually EfficientNet [9]reached a GAP value of 0.1423. On the whole, the gap between ResNet (average 0.0845) and EficientNet (average 0.08805) is not very large, which means that in the first 10 epochs, the convergence of ResNet and EfficientNet [9]is similar. From the perspective of network stability, ResNet [5] is more stable than EfficientNet [9], and there is no particularly large fluctuation. In contrast, EficientNet [9] has certain fluctuations in the fourth epoch. Analyzing from the network architecture, EficientNet [9] is more plastic for convolutional neural networks, and its convergence speed is faster. At the same time, according to the final result, EfficientNet [9]is more accurate than ResNet [5], so we choose to use EfficientNet

epoch	1	2	3	4	5	6	7	8	9	10
ResNet18	0.004	0.0309	0.049	0.0782	0.0929	0.1019	0.113	0.1194	0.1316	0.1336
EfficientNet	0.0359	0.0569	0.0582	0.0541	0.0864	0.0976	0.1056	0.1097	0.1338	0.1423

Table 1. Table represent the 10 epochs GAP values of different backbone

[9] as our backbone.

Then, due to the complexity of the landmark, we added ArcFace [2] as an additional compensation in the process of calculating the loss. ArcFace [2] directly maximizes the classification boundary in the angular space, which theoretically has an effect on the buildings in the landmark. After 10 epoch training and testing, the results are shown in the figure below5. The red curve and blue curve are ResNet and EfficientNet [9]mentioned above, and the gray curve is ArcFace+Efficientnet [9]. It can be seen from the image that after adding ArcFace as the extra value of the loss function, the result has a significant increase. At the same time, comparing the first four epochs with EfficientNet [9], we can find that after adding ArcFace as a loss function, the curve result does not fluctuate. We guess that it is the result of the instability of training. Due to time reasons, we did not conduct multiple training comparison results for EfficientNet [9]and Resnet, but we referred to the players in this competition on Kaggle. They used ResNet50+ArcFace to train same dataset for 16 epochs. The result in validation test sample is about 0.20. It can be seen that EfficientNet [9]still obtains a higher result than ResNet even when the number of training times is low. In the end, the result of ArcFace+EfficientNet we chose showed a GAP value of 0.2102, which was an increase of nearly 50% compared to using EfficientNet [9]alone.

Finally, we add the non-landmark classifier mentioned above to our model architecture. The image will first go through the non-landmark classifier, and then select the multiple landmark classifier according to the number of pictures in the class. The result will be obtained in the end. After the above architecture, we also trained 10 epochs. The final GAP value of 0.2423 is obtained. Due to the particularity of the dataset, non-landmark images only exist in the test dataset, and there are no non-landmark images in our training set. As a result, it is difficult for us to classify non-landmark separately. Because the model cannot judge situations that have never been trained.The overall network model results are shown in the table2. After 10 epochs of training, Non-landmark+ArcFace+EfficientNet got the best result, and its GAP value reached 0.2423. Compared with using EfficientNet alone, it increased by 70%.

Network	GAP	Loss
ResNet	0.1336	3.7778
EfficientNet	0.1423	3.8902
ArcFace+EfficientNet	0.2102	2.5897
Non-landmark+ArcFace+EfficientNet	0.2423	

Table 2. Table represent the 10 epochs GAP values of different backbone

5. Conclusion

Due to time reasons, we cannot use the controlled variable method to test experiments for all models, but according to our selection and analysis, the classification of landmarks also has certain results. First of all, for ResNet, EfficientNet does not have a particularly prominent advantage in its performance in this project. We estimate that the number of training times of 10 epochs does not make EfficientNet have a good training effect. Because the data types in this dataset are mottled and complicated, and the number of classes is huge, more training times are needed to support. Second, the advantage of EfficientNet is that it automatically expands in three dimensions, so in the case of a huge sample size, sufficient training support is required. Secondly, by adding ArcFace loss, we greatly increased the convergence speed of the network, with a nearly 50% improvement. This shows that ArcFace is not limited to facial recognition projects. Because ArcFace discriminates the angle space, it also has a very good effect on some buildings. From an extended point of view, ArcFace has a wide range of applications. Some models that are not sensitive to angles can be added to ArcFace to perform loss calculations to achieve better results. Finally, we proposed the attention theory. A non-landmark classifier is added to the model. On the whole, its accuracy has been improved by a certain amount, but it is not particularly obvious. We estimate that it is because the main features of landmark are too many and not obvious. Therefore, the performance of non-landmark in the validation set is not satisfactory.

Finally, as an extended study of this project, we only set the threshold of 30 and 15 in a limited time, and generated two classifiers (multiple landmark classifiers). Since the process of dividing the threshold does not carry out multiple tests, in the future learning process, we will increase the number of classifiers according to the actual situation. At the same time, we will set different thresholds to perform training process and choose the thresholds experimentally.

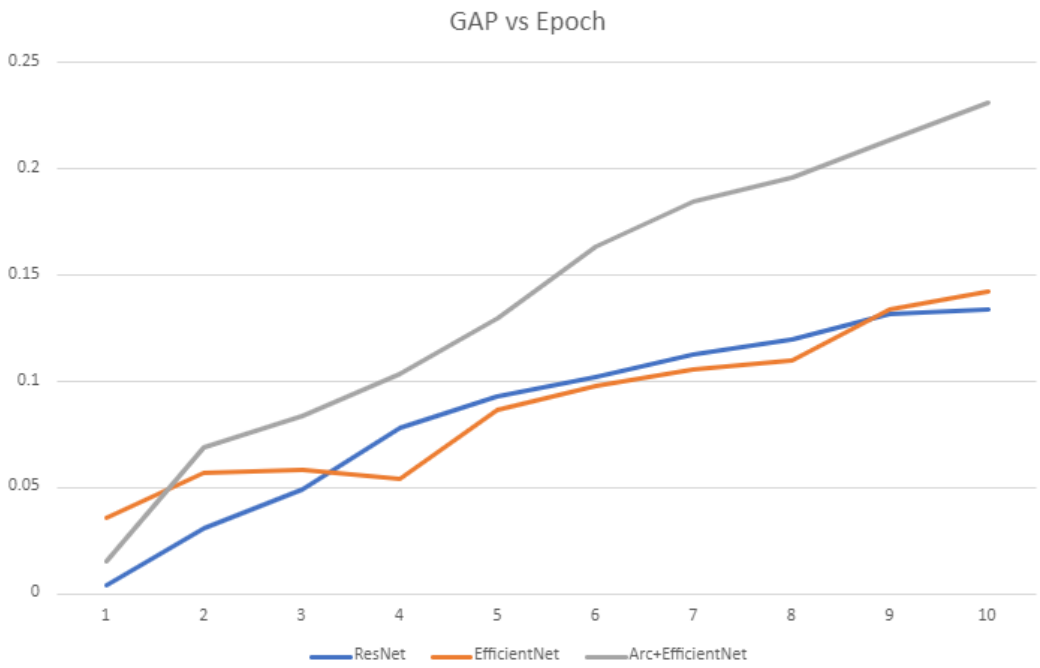

Figure 5. GAP vs epoch with different network architecture.

References

[1] chankhavu, 2020 (accessed Oct, 2020).

[2] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2019.

[3] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[4] Yinzheng Gu and Chuanpeng Li. Team jl solution to google landmark recognition 2019, 2019.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[6] Christof Henkel and Philipp Singer. Supporting large-scale image recognition with out-of-domain samples, 2020.

[7] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[8] Mingxing Tan and Quoc Le. EfficientNet: Rethinking model scaling for convolutional neural networks. volume 97 of *Proceedings of Machine Learning Research*, pages 6105–6114, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

[9] Mingxing Tan and Quoc V Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *arXiv preprint arXiv:1905.11946*, 2019.

[10] Tobias Weyand, Andre Araujo, Bingyi Cao, and Jack Sim. Google landmarks dataset v2-a large-scale benchmark for instance-level recognition and retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2575–2584, 2020.

[11] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2017.