Landmark Recognition

https://youtu.be/DorCOLR_yOE

Renke Cai
CS231N
Renke.cai@stanford.edu

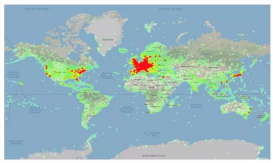
Chenjiao Wang
CS230
Chenjiao.wang@stanford.edu

Motivation:

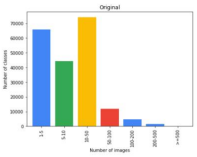
The landmark recognition problem comes from a Kaggle Challenge launched by Google Inc. in April, 2019. The goal is to build models that recognize the correct landmark (if any) in a large-scale dataset of images. This topic is of particular interest to all due to its applications in daily life -- for example, the visual search, we want to know the presence of a specific landmark from the photo; we could do the personal photo search to tag the trips we ever made; the landmark recognition is also widely used in the visual games such as Pokeman and other virtual reality games.

Data Features

The dataset was constructed by clustering images based on geo-locations. The original training dataset (over 500 GB) contains 4,132,914 images of 203,094 famous as well as not so famous landmarks all over the world.



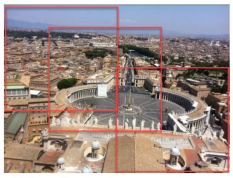
The numbers of images varies high from classes to classes. Maximum images in a class are 10247 while a number of other classes contain only 1 pictures



In the time of interest, we filtered out 6,401 classes which consist of 1,243,915 images in total for our training.

Data Augmentation

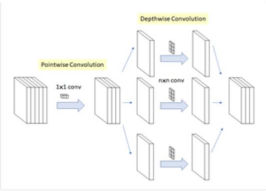
we customized an image cropping function that crop the original image into resolution of 150 * 150. Note that the original image we prepared are of width 256 while height is set so that original scale is preserved.



- Cropping probability: the probability of whether to perform image cropping or not;
- Image resizing scale: resizing the original image into this size before cropping;
- Way of random cropping: cropping from the center or corners or edges.

Xception Model

Xception takes one step further than Inception v3. Instead of partitioning input data into several compressed chunks, it maps the spatial correlations for each output channel separately, and then performs a 1x1 depth-wise convolution to capture cross-channel correlation.



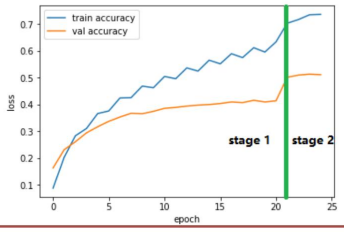
Base Model	Xception Model 1	Xception Model 2
Top-layer Model	Generalized mean pool Dropout Output(softmax)	Generalized mean pool Dropout FC(dim=256,Relu) Dropout Output(softmax)
Frozen Layers	Bottom 80 conv layers	Bottom 20 conv layers
Learning Rate	0.0001	0.00001
Optimizer	Adam	Adam
Loss function	Categorical Cross Entropy	Categorical Cross Entropy
Cropping probability	0.1 to 0.5 linear increasing by epoch	0.8 to 1.0 linear increasing by epoch
Resizing scale	Resizing to 180*180 before cropping	Resizing to 229*229 before cropping
Way of cropping	Random center position	Random center position

Results

	Accuracy	GAP
Training set	0.7496	--
Validation set	0.5111	--
Test set	0.5159	0.4709
Test set with voting mechanism	0.5389	0.5964

Predict through Voting:

Confidence score is then based on the weighted average probability of the majority label (i.e. the final prediction).

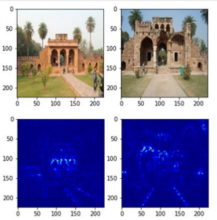


Visualization:

For a better visualization, we employed the method of guided saliency.

We use these gradients to highlight input regions that cause the most change in the output. Intuitively this should highlight salient image regions that most contribute towards the output.

$$\frac{\partial output}{\partial input}$$



Conclusion

The quantitative and qualitative analysis of our trained model indicates a satisfactory performance on the 6k+ classes of landmarks.

Data augmentation especially image cropping played a significant role in improving the accuracy. Also cropping at test time and use the majority win method gives better prediction accuracy, though it is heuristic.

Future Works

Once the problem is expanded to the original dataset with 200K+labels, it could be way more difficult to train a classifier using a pure deep learning architecture. Some hybrid method includes unsupervised learning such as KNN could be powerful.

Local feature descriptor for large-scale image retrieval called DeLF could be useful for this particular challenge. It extracts local features from images and matches them. We used it for matching local features of test images to images known to be landmarks.