MIE 1516 Final Project

Application of Attention Mechanism in Fine-grained Image Classification

Siyu Chen

1001181945

1. Project Introduction and Motivation

Although CNN has exhibited good performance on traditional image classification tasks, such as MNIST, CIFAR, there are still challenges remaining in classifying very similar images correctly. For example, CNN have difficulties in identifying the cats and the dogs or the birds and the planes in CIFAR-10 dataset, indicated by the large value in the cat-dog or bird-plane cells in the confusion matrix. It is even more difficult for a neural network to identify images that average human beings without specific knowledge could identify correctly, such as identifying the species of a bird or the brand of a car. Such task is known as fine-grained image classification and there are lots of efforts on developing better architectures in the recent years.

As a result, this project is aimed to explore the ability of deep neural network on fine-grained image classification, i.e. identify subclasses from similar images. It is a hot topic in recent years in image classification. Moreover, not only the researchers but also the average people will gain lots of benefits from this technique. For example, people can get more information or know what kind of keywords they need to put into a search engine more specifically if the neural network tell them the subclass of a object (such as a BMW X3 2018, orchid) rather than a general class (such as a car/SUV, flower).

2. Attention Mechanism

One of the innovative techniques developed in CNN is the attention mechanism. The output of a 2D-convolution layer can be regarded as a feature map, a representation of the original images in different dimensions. From these feature map, we can extract the attention, the area of the pixels that the neural network is trying to look at [1]. These attentions can help the neural network to look closer at the fine features of the object in an image, which then helps to improve the performance of neural network.

3. Architectures and results

This project gradually improves a CNN network on the Stanford Car data set, which was used in FGCcomp in 2013 [2]. There are 8144 labeled data in total. 80% of the data are used for training and the rest 20% of data are used for testing. For the train data set and test data set respectively, the images are zero-centered for each channel. The Five architectures are tried out and their results are compared. For all the architectures, RMSProp with a learning rate 0.0001 is used to train the network. The batch size is set to 32 due to the limitation of memory.

(1) The first and simplest architecture is the CNN used in assignment 3 with slightly modification, which achieved ~82% accuracy on the test dataset of CIFAR-10. The architecture is shown in Figure 1. As shown in Figure 2, after 300 epochs, the accuracy for the train set is 99.91% but the accuracy for the validation set is only 2.58%. The architecture is extremely overfitted and the neural network cannot differentiate the difference between these similar images for test dataset although it performs well in CIFAR-10.

Figure 1. The simple CNN architecture

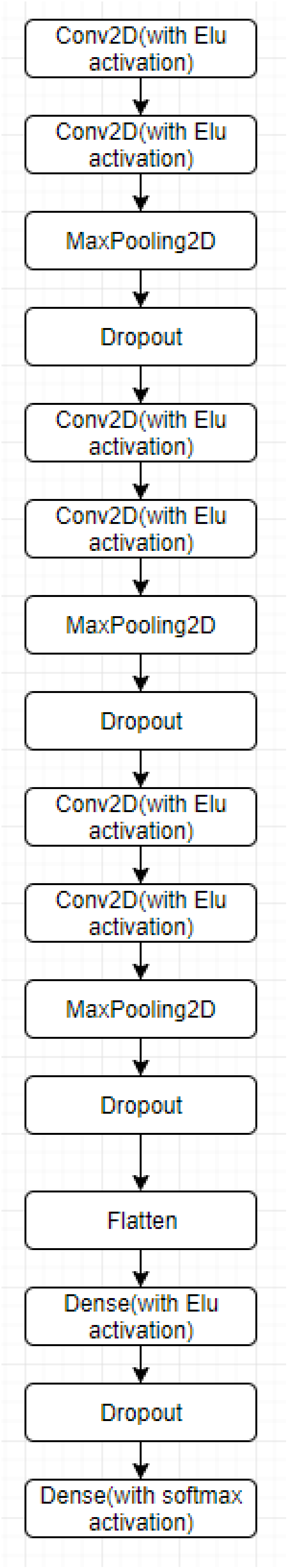
Conv2D(with Elu activation)
Conv2D(with Elu activation)
MaxPooling2D
Dropout
Conv2D(with Elu activation)
Conv2D(with Elu activation)
MaxPooling2D
Dropout
Conv2D(with Elu activation)
Conv2D(with Elu activation)
MaxPooling2D
Dropout
Flatten
Dense(with Elu activation)
Dropout
Dense(with softmax activation)

Figure 1. The simple CNN architecture

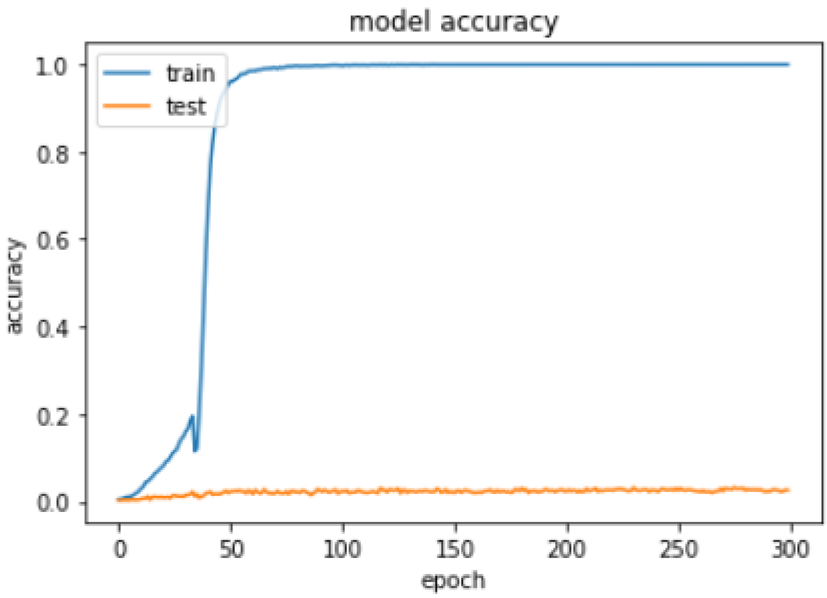

Figure 2. The accuracy versus epoch over 300 epochs

(2) In the second model, A pre-trained VGG-19 model is used for fine tuning. All the layers other than the last 4 fully-connected layers are frozen. The fully connected layers are dropped and customized fully-connected layers are added. The architectures are shown in Figure 3. As shown in Figure 4, fine-tuning a pre-trained model is slow in that both curves are just about to converge after 300 epochs. The accuracy for train group is 89.1% and that for test group is 24.55%. The fine-tuned model performs better than the previous one since it is pre-trained with lots of data. The model reduce the effect overfitting and is able to detect some of the minute differences among subclasses.

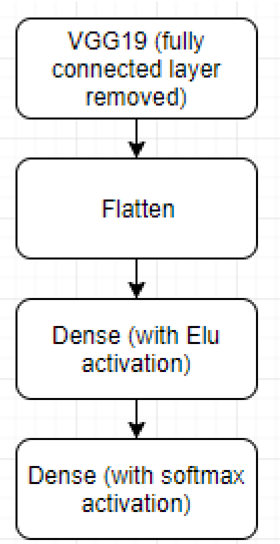
VGG19 (fully connected layer removed)
Flatten
Dense (with Elu activation)
Dense (with softmax activation)

Figure 3. The architecture for fine-tuned VGG19 architecture

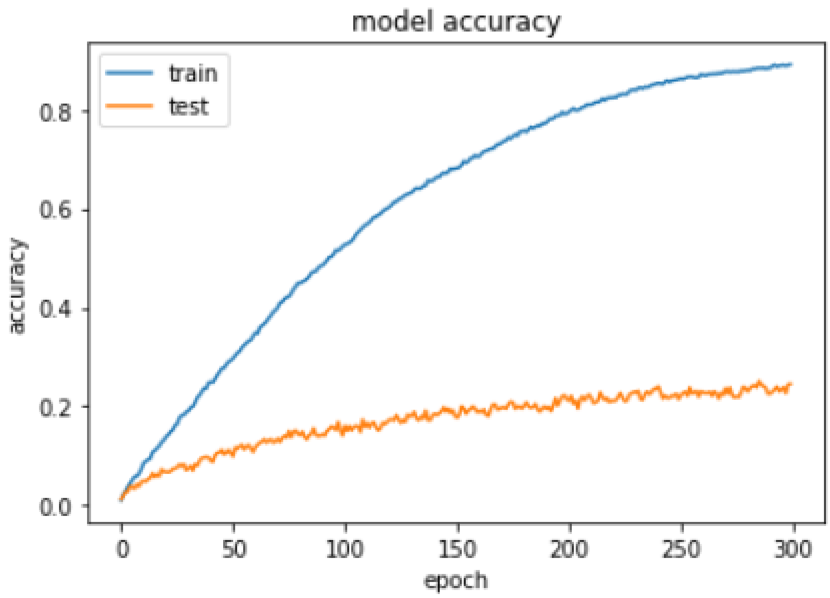

Figure 4. The accuracy versus epoch over 300 epochs

(3) In this architecture, the pre-trained VGG19 model is used with the last 4 fully-connected layers removed. The output of this VGG19 model of the last MaxPooling2D layers is a tensor of size (7,7,512) for each input image. It has 512 channels of 7*7 feature map representing the original images. According to B.Zhou et.al [1], the global average pooling (GAP) of these feature maps gives the weights, i.e. the relative importance of different channels. This layer is followed by a fully-connected to give the output class. The attention activation map for each class is the weighted sum of all these channels. The model is shown in Figure 5.

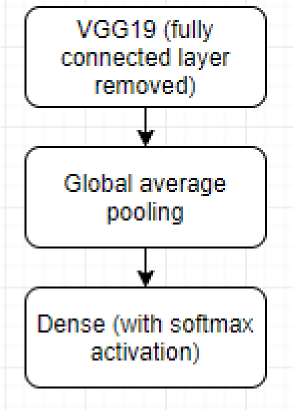
VGG19 (fully connected layer removed)
Global average pooling
Dense (with softmax activation)

Figure 5. The architecture for fine-tuned VGG19 model with attention activation

The accuracy at each curve is shown in Figure 6. After 200 epochs, the accuracy for training dataset is 98.65% and the accuracy for the testing dataset is 32.84%. Moreover, this architecture is converging much faster than the previous architecture in that it converged after around 150 epochs. By simply adding the attention activation (GAP layer), the neural network is able to learn the subclass difference much better and more efficiently.

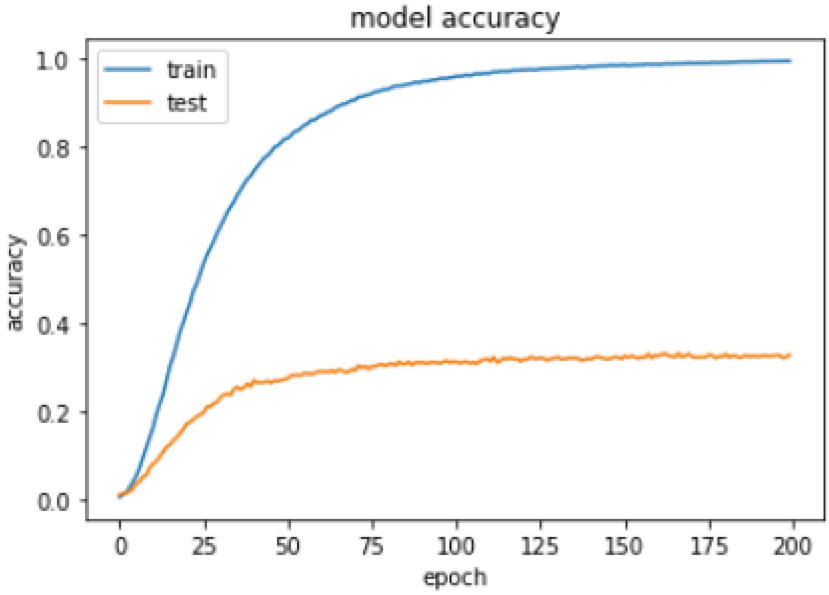

Figure 6. The accuracy versus epoch over 200 epochs

Attention map are stacked with the original images to visualize where the neural network is looking at. The images are shown in Figure 7. The bluer region indicates the places where the neural network is looking at. By using GAP, the neural network can focus on exactly the region where the cars are and the component of a car, such as the wheel, front window, etc. The pictures on the left column shows the attention activation map of the labelled category while the pictures on the right column shows the attention activation map of the predicted category. It can be shown that the labelled category and predicted category have similar attention activation map for images that are incorrectly classified (first, second and fifth image). Moreover, in these incorrectly classified images, the neural network seems to be distracted on irrelevant features, such as the words on a advertisements, or other objects in the background of a photo.

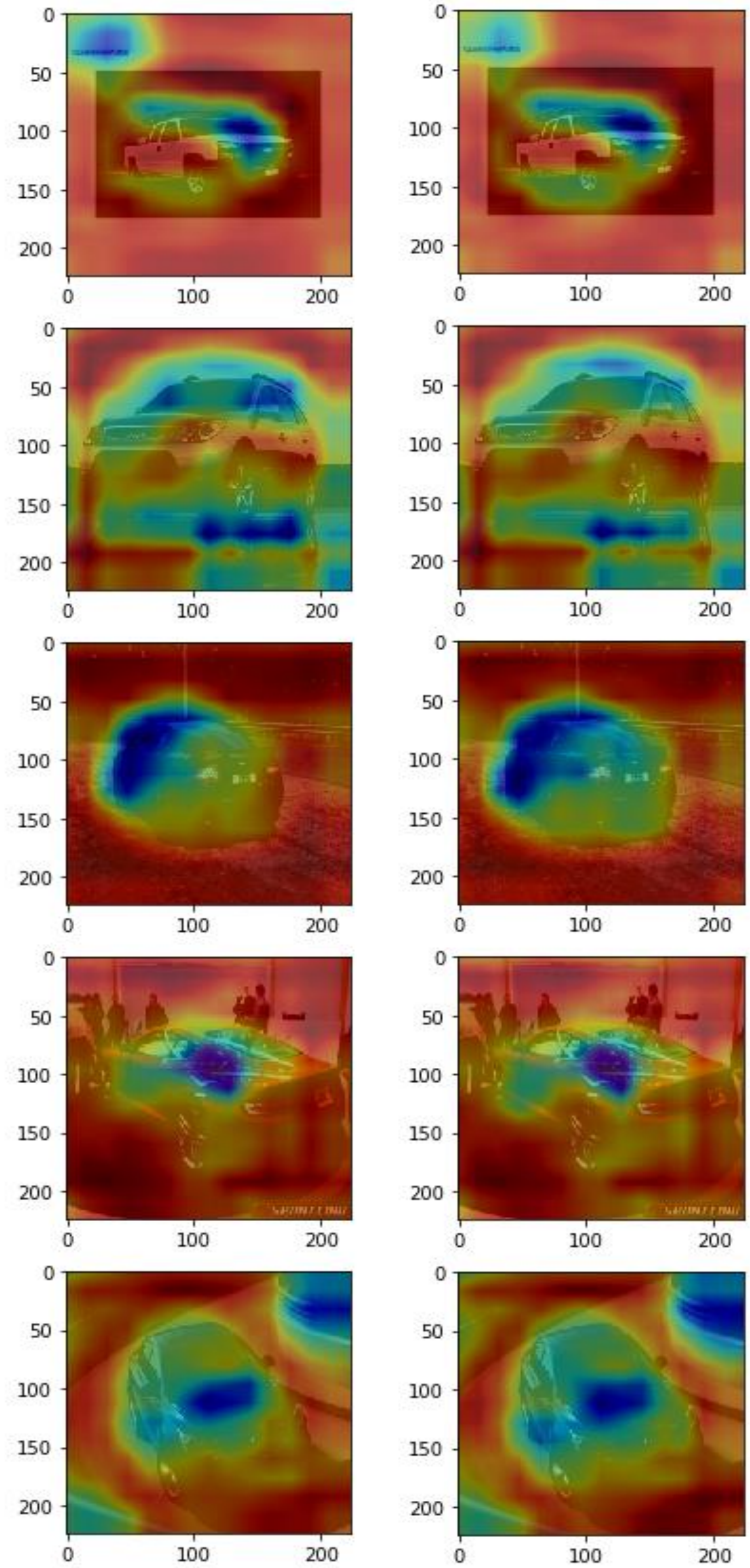

Figure 7. Attention map shown on the original images on test data. The left series shows the attention map for labelled category and the right series show the attention map for predicted category.

(4) In this architecture, attention map is used for data augmentation. The architecture is shown in Figure 8. The image is fed into the VGG19 to generate feature map F^{H*W*C} with height H, width W and C channels. For each pixel on the feature map, the value is averaged along the channel axis to get the one-channel attention map A^{H*W*1}. This attention map is then duplicated to 3 channels and unsampled to the size of the input images. The new images are generated by applying an element-wise multiplication between the input image and the attention map. The original image and the augmented images are then used to train the network.

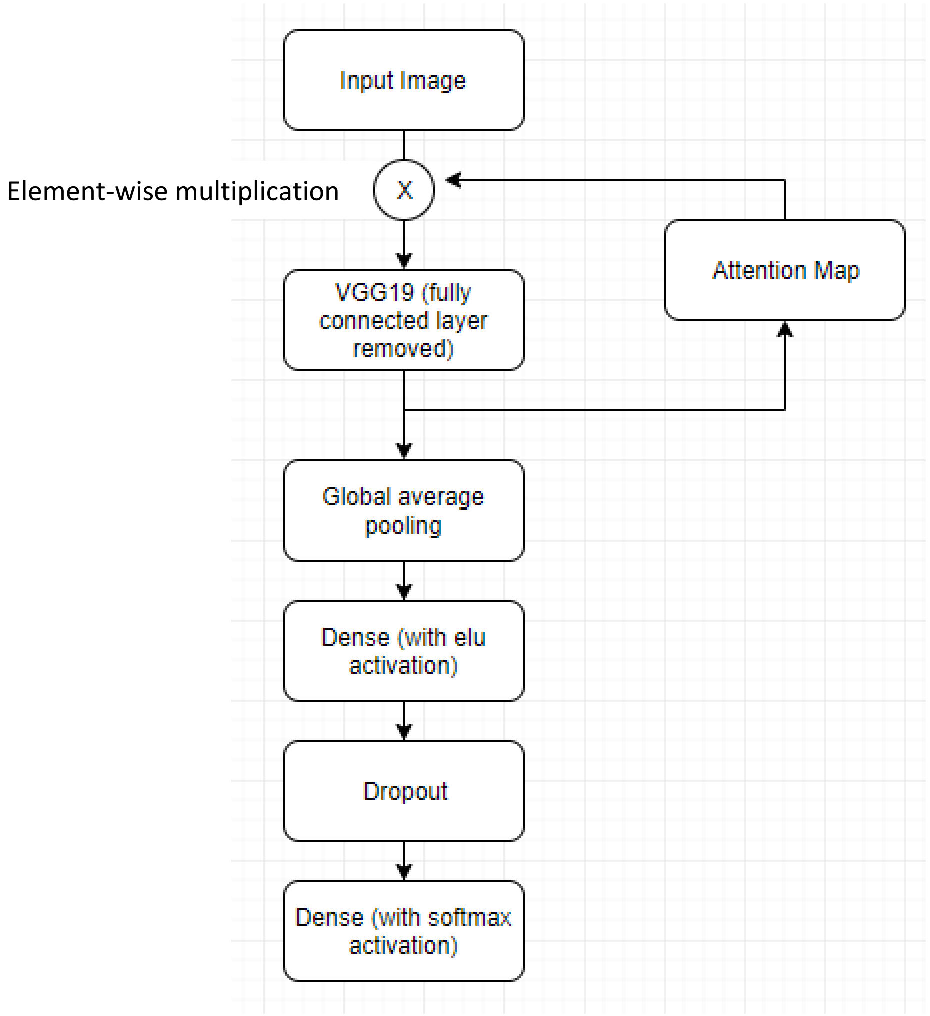
Input Image
Element-wise multiplication
X
Attention Map
VGG19 (fully connected layer removed)
Global average pooling
Dense (with elu activation)
Dropout
Dense (with softmax activation)

Figure 8. The architecture of fine-tuned VGG19 with attention activation and augmentation

The accuracy at each epoch is shown in Figure 9. After 150 epoch, the architecture reached an accuracy of 99.02% for train data and 22.04% for the test data. The architecture converged slower than the previous architecture because of the augmented data added.

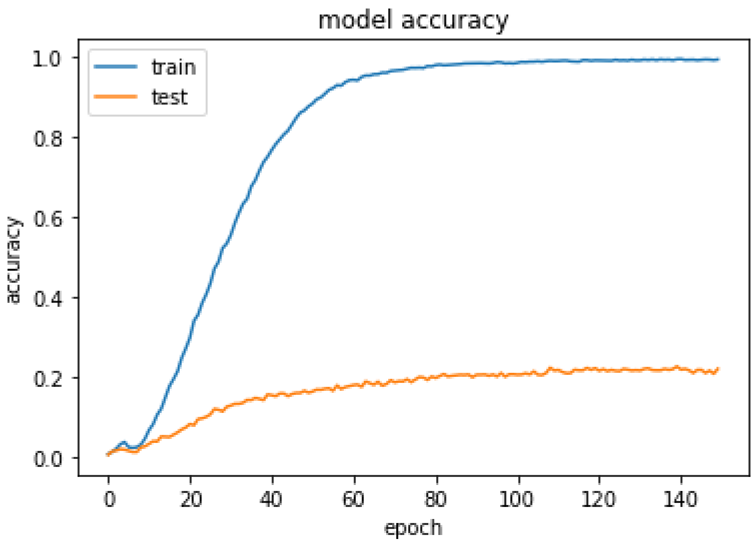

Figure 9. The accuracy versus epoch over 150 epochs

(5) Another way of applying attention map in neural network is tried as well. The weights of attention activation are applied directly on the feature map as inspired from the work of Sanghyun et. al [3]. The architecture is shown as in Figure 10. The new feature map is the weighted sum of all the channels, where the weights are obtained by applying GAP on the original feature map. All the feature maps are used to train the neural network. After 300 epochs, the neural network reached a 99.52% accuracy on the training dataset and 23.63% accuracy on test dataset, showing no obvious improvement compared to the previous architecture.

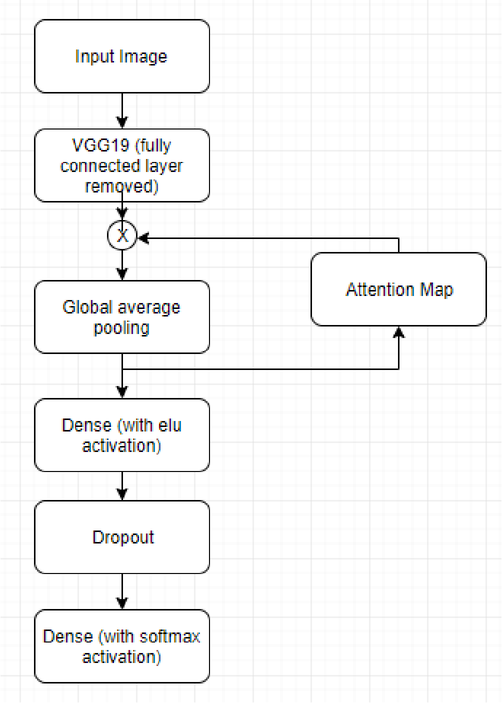
Input Image
VGG19 (fully connected layer removed)
X
Attention Map
Global average pooling
Dense (with elu activation)
Dropout
Dense (with softmax activation)

Figure 10. The architecture of attention augmentation on feature map

4. Conclusion

The results of all the architectures are listed in Table 1. By introducing the attention mechanism to neural network like VGG19, the neural network is converging much faster and have higher accuracy on test dataset than the original CNN on fine-grained image classification. The attention map shows the critical region where the neural network is looking at and contains differences between similar sub-classes. The data augmentation with attention have a similar effects but is not as ideal as I expected. The reason may lie in the neural network is not deep and wide enough compared to those ones described in the paper and the hyperparameters such as learning rate and batch size are not fully tuned due to the limitation of computational resources and time of this project. There are more techniques on attention image augmentation can be implemented, such as the data cropping with attention and data dropping with attention as discussed in the work of Tao et.al [4]. The future work will be emphasised on training a deeper and wider neural net with more advanced attention augmentation technique to see if it improves the performance of neural network.

Table 1. Comparison of accuracy on dataset for different architecture.

Architecture	Accuracy on test dataset
Simple CNN	2.58%
Fine-tuning VGG19	24.55%
Fine-tuning with attention activation	**32.84%**
Fine-tuning with attention augmentation	22.03%
Attention augmentation on Feature Map	23.63%

5. References

[1] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning Deep Features for Discriminative Localization. CVPR'16 (arXiv:1512.04150, 2015).

[2] Jonathan Krause, Michael Stark, Jia Deng, Li Fei-Fei. 3D Object Representations for Fine-Grained Categorization. 4th IEEE Workshop on 3D Representation and Recognition, at ICCV 2013 (3dRR-13). Sydney, Australia. Dec. 8, 2013.

[3] Sanghyun Woo, Jongchan Park, Joon-Young Lee, In So Kweon. CBAM: Convolutional Block Attention Module. ECCV 2018(arXiv:1807.06521)

[4] Tao Hu, Honggang Qi. See Better Before Looking Closer: Weakly Supervised Data Augmentation Network for Fine-Grained Visual Classification.